3
T57
[illegible]

The Investigation of
Service Time Distributions

by

Eleanor A. Lester

Technical Report CSRG - 25

April 1973

COMPUTER SYSTEMS RESEARCH GROUP
UNIVERSITY OF TORONTO

$2.00

The Investigation of Service Time Distributions

by

Eleanor A. Lester

Technical Report CSRG - 25

April 1973

The Computer Systems Research Group (CSRG) is an interdisciplinary group formed to conduct research and development relevant to computer systems and their application. It is jointly administered by the Department of Electrical Engineering and the Department of Computer Science of the University of Toronto, and is supported in part by the National Research Council of Canada.

ABSTRACT

In the absence of full information regarding service times of jobs in a system, analysis and simulation studies have shown that performance can be improved by using whatever predictive information is available. Empirical studies of actual systems have found that the overall service time distribution of jobs is highly skewed with a very long tail. However, algorithms which use distribution information can make better use of distributions with lower variance. In this thesis, we investigate the effects of conditioning service time distributions on characteristics of jobs known before the jobs are processed. The resulting conditional distributions are compared to determine which characteristics are the best predictors of actual service time. We attempt to fit these distributions to known theoretical distributions. The implications of the empirical distributions and their associated mean residual life and rank functions in the choice of a scheduling algorithm are considered.

ACKNOWLEDGEMENTS

I would like to thank Professor K. Sevcik for his valuable guidance and constant encouragement in the preparation of my thesis. I would also like to express my appreciation to Professor D. Wortman for his useful suggestions and criticisms.

I would like to thank Dr. D. McNaughton, Mr. J. Lees, and Mr. P. Young of the University of Toronto Computer Centre for their cooperation and assistance in obtaining data and operating system information for this study.

I would like to thank my mother for her understanding and assistance in my endeavors.

This research was supported by the Defense Research Board under grant DRB 9931-34. Financial assistance was also gratefully received from the Department of Computer Science for my first academic year here.

TABLE OF CONTENTS

Page

Abstract i

Acknowledgements ii

Contents iii

Chapter I

- Introduction 1
- Definitions 1
- Source of the Problem 10
- Types of Disciplines Formulated 12
- Analytical Studies 13
  - Assumptions and Performance Measures of Scheduling Models 13
  - Optimality of Scheduling Disciplines 14
- Comparative Performance of Optimal or Good Scheduling Discplines 17
- Service Time Distribution Studies 19
- Implications of Empirical Studies 23

Chapter II

- Objectives of Our Investigation 27
- Methods Employed in the Processing of the Data 27
  - Distribution Fitting 28
  - Measurements of Differences in Empirical Distributions 32
- Data Gathering and Handling 32
  - Source of the Data 33
  - Description of the Data Gathered 34
  - Program Description 36
  - Some Implementation Considerations of the Program 39
  - Program Output 40

Chapter III

- Experiments and Results 43
  - Representative Distributions 43
  - Best Partitions 49
    - Conditioning on One Parameter 49
    - Conditioning on Two Parameters 54
  - Observations on the Empirical Distributions 55

Digitized by the Internet Archive
in 2018 with funding from
University of Toronto

https://archive.org/details/technicalreportc25univ

Zero Parameter Conditioning 56
One Parameter Conditioning 57
Two Parameter Conditioning 61
Distribution Fitting 68
Distribution Characterizations 72

Chapter IV

Conclusions 73
Best Partitioning Parameters 73
Distribution Fit 77
Implications in the Scheduling Discipline Area 78

References 80

CHAPTER I

Introduction

In our modern society, an increasing amount of our workload is being handled by computers. Because of the cost of using computers, much research has dwelt on the problem of producing efficient computer systems. An important area in this regard is the scheduling of the processing of jobs that have arrived at a system. This scheduling attempts to maximize the utilization of computer resources to provide fast turnaround to the user, at minimum cost. However, scheduling disciplines being used in actual systems may not be optimal. Before a new scheduling discipline can replace an existing one that performs reasonably well, much analytical and empirical study must be centred on the assumptions of the discipline and its proposed capabilities. Empirical investigation into a particular area of this subject is the intention of this study.

Before elaborating on this area, we present some definitions relevant to our topic and necessary to readers interested in this study but unfamiliar with the terminology.

Definitions

We present three general definitions followed by (a) scheduling rules, (b) associated functions, (c) distributions, and (d) classifications of distributions.

(1) Scheduling discipline: a method used to determine the order in which jobs are processed in a system. This order can be determined through a selection rule (selection discipline) which is dependent upon the value of a function involving attributes of an individual job (request).

(2) Optimal scheduling discipline: a scheduling discipline which guarantees the most desirable value of some particular measure of performance.

(3) Preemption: the interruption of the processing of a job before its completion. The processing of the job may be continued later from its point of progress at the time of preemption.

A. Scheduling Disciplines

(1) FIFO: first in, first out. The job arriving first is processed first.

(2) LIFO: last in, first out. The job arriving last is processed first.

(3) PLIFO: preemptive last in, first out. The job arriving last is given highest priority and preempts a job

currently in service.

(4) RAND: random selection. The job to be processed is selected at random from those waiting for service.

(5) RR: round robin. Jobs waiting for service are queued. The job at the head of a queue is processed for a quantum of time. If it does not complete within that quantum, it joins the end of the queue. There are a number of variations of RR:

(a) SQRR: single quantum round robin. Each job receives a set quantum as described above.

(b) MQRR: multiple quantum round robin. A job is processed until completion or until it receives one full quantum during which no jobs arrive. If it has not completed in the latter case, it joins the end of the queue as in SQRR.

(c) CYCRR: cycle oriented round robin. A job is processed for a quantum length which is inversely proportional to the number of jobs waiting in the queue at the instant the time is allocated.

(6) FBn: feedback to n levels. A job on arrival is assigned to a highest priority queue. The job selected for service is chosen from the highest priority non-empty queue, and is serviced for a quantum. If it does not complete within that time, it joins the end of the queue with the next lower priority.

(7) PS: processor sharing. This is similar to RR,

but with the quantum length tending to zero. The processor is divided equally among all jobs in the system.

(8) PPS: preemptive processor sharing. This is like FBn with the quantum length tending to zero at every queue level.

(9) SPT: shortest processing time. The job with the shortest processing time is serviced to completion.

(10) SRPT: shortest remaining processing time. The job with the shortest remaining processing time has highest priority. If a job arrives whose processing time is shorter than the remaining processing time of the job currently in service, the arriving job preempts and is serviced.

(11) SEPT: shortest expected processing time. The job with the shortest expected processing time is processed to completion.

(12) SERPT: shortest expected remaining processing time. This is the same as SRPT except that the expected remaining processing time is used rather than the actual remaining processing time.

(13) SR: smallest rank. The job with current smallest rank is given highest priority. The rank of a job is calculated from the service time distribution of that job, as a function of attained service. Interval rank is the ratio of A to B where A is the expected investment of processor time in allowing a job up to e more units after it has attained t units of service without completing, and B is

the linear loss rate times the expected payoff in allowing a job up to e more units of processor time. The expected payoff is the probability that the job will complete within the next e units of service. The rank of a job is defined as the minimum interval rank for a quantum of any length e (Sevcik 1971). This discipline assumes that an arriving job will belong to a particular request class with a processing time distribution different from those of other request classes. The request class is determined from the characteristics of the arriving job. For example, jobs requiring 60K of core may have a distribution different from those jobs requiring 130K of core.

E. Associated Functions

Associated with any job in a system is a loss function or waiting cost function. It indicates the decrease in the value of service given a job, from an immediate completion to completion after a certain time interval. Loss functions are dependent upon the cost of storing information relevant to a request being serviced, upon external priorities, upon the usefulness of the result, and upon other external factors. If information created during the servicing of a job can be retained at negligible cost, its loss function is monotone non-decreasing. Three types of loss functions are:

(1) linear. The loss is proportional to the time of the job in the system.

(2) convex. The second derivative of the loss function is nonnegative.

(3) concave. The second derivative of the loss function is nonpositive.

Scheduling disciplines can be classified according to the loss functions they presume. SPT/c assumes a linear loss function is associated with each job. It is different from SPT in that in choosing the highest priority job to be serviced, the smallest ratio of the processing time of a job to its loss function value, c, is considered, rather than the shortest processing time. SPRT/c, SEPT/c, SERPT/c are defined similarly.

C. Distributions

Other factors influencing scheduling disciplines are interarrival distributions and processing distributions. These distributions may assume various shapes. Below are given the density and cumulative distribution functions of the theoretical distributions considered in this study. All of the functions have values of zero when t is negative. Figure 1 gives the general shape of the distributions.

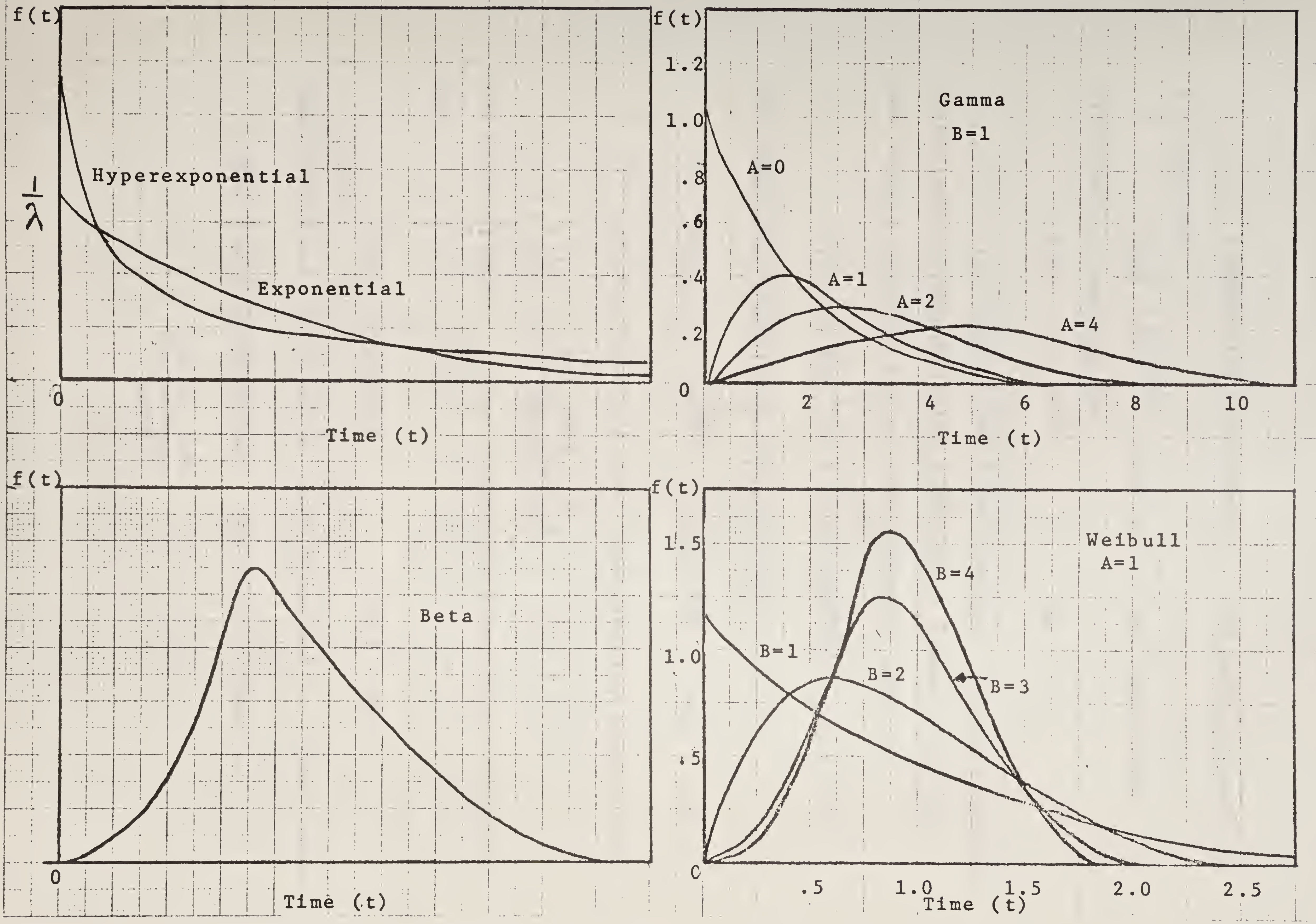


FIGURE 1. DENSITY FUNCTIONS OF THEORETICAL DISTRIBUTIONS

(1) Exponential distribution. Its density function is given as:

$$f(t) = \lambda e^{-\lambda t} \qquad \text{for } t \geq 0, \text{ and cumulative}$$

distribution function as:

$$F(t) = 1 - e^{-\lambda t} \qquad \text{for } t \geq 0.$$

(2) Hyperexponential distribution. This distribution is the sum of exponentially distributed random variables. The density function for the two parameter case is:

$$f(t) = p\lambda_1 e^{-\lambda_1 t} + (1-p)(\lambda_2 e^{-\lambda_2 t}) \qquad \text{for } t \geq 0,$$

The cumulative distribution function is:

$$F(t) = 1 - pe^{-\lambda_1 t} - (1-p)e^{-\lambda_2 t} \qquad \text{for } t \geq 0.$$

(3) Erlang-k distribution. The density function is:

$$f(t) = \frac{(t^{k-1} e^{-k\lambda t} (k\lambda)^k)}{(k-1)!} \qquad \text{for } t \geq 0,$$

and k an integer ≥ 1. The cumulative distribution function is:

$$F(t) = \int_0^t f(x)\, dx \qquad \text{for } t \geq 0.$$

This function is too complex to be expressed in closed form.

(4) Gamma distribution. The density function is:

$$f(t) = \frac{t^{(A-1)} e^{-t/B}}{\Gamma(A)\, B^A} \qquad \text{for } t \geq 0$$

$$\text{where } \Gamma(A) = \int_0^\infty t^{A-1} e^{-t}\, dt$$

$$\text{for all } A > 0.$$

The cumulative distribution function is:

$$F(t) = \int_0^t f(x)\, dx \qquad \text{for } t \geq 0.$$

The exponential distribution is a special case of the gamma distribution when $A = 1$. The Erlang-k distribution is also a special case of the gamma distribution where A is an integer ≥ 1 ($k = A$, $\lambda = 1/(AB)$ where k and λ are the parameters of the Erlang-k distribution).

(5) Weibull distribution. The density function is:

$$f(t) = (A/B)\, e^{-(t/B)^A} (t/B)^{A-1} \text{ if } t \geq 0,\ A > 0,\ B > 0.$$

The cumulative distribution function is:

$$F(t) = 1 - e^{-(t/B)^{A-1}} \quad \text{if } t \geq 0,\ A > 0,\ B > 0.$$

The exponential distribution is a special case of the Weibull distribution when $A = 1$.

(6) Lognormal distribution. A random variable $Y = e^X$ in which X is normally distributed with parameters m and σ is said to have a lognormal distribution with parameters m and σ.

(7) Beta distribution. The density function is:

$$f(t) = (\Gamma(A+B))/(\Gamma(A)\Gamma(B))\,(t^{A-1})\,(1-t)^{B-1}$$

$$\text{for } 0 \le t \le 1.$$

The cumulative distribution function is:

$$F(t) = \int_0^t f(x)\,dx \qquad \text{for } 0 \le t \le 1.$$

The density and cumulative distribution functions have value zero when $t \ge 1$. It should be noted that the Beta distribution is finite.

D. Distribution Classifications

Service time distributions can be characterized by:

(1) Failure Rate(FR). This is the probability that a job with a certain amount of attained service will complete in the next instant. Failure rate can be:

(a) increasing, e.g., FR of an Erlang-k distribution,

(b) decreasing, e.g., FR of a hyperexponential or a Weibull distribution,

(c) constant, e.g., FR of an exponential distribution.

(See Figure 2.) It is defined as:

$FR(t) = f(t)/(1-F(t))$ where $f(t)$ and $F(t)$ are the density function and cumulative distribution function, respectively.

(2) Mean Residual Life (MRL). This is the expected remaining service time of a job when it has attained a certain amount of service, defined as:

$$MRL(t) = (\int_t^\infty (x-t)f(x)dx)/(\int_t^\infty f(x)dx)$$

where $f(x)$ is the distribution function. It may be:

(a) increasing, e.g., MRL of a hyperexponential or a Weibull distribution,

(b) decreasing, e.g., MRL of an Erlang-k distribution,

(c) constant, e.g., MRL of an exponential distribution.

(3) Rank. Its dynamics are the same as those for MRL.

Source of the Problem

Concern over the manner in which jobs pass through a computer system has led to the development of a number of scheduling disciplines. Before the concepts of

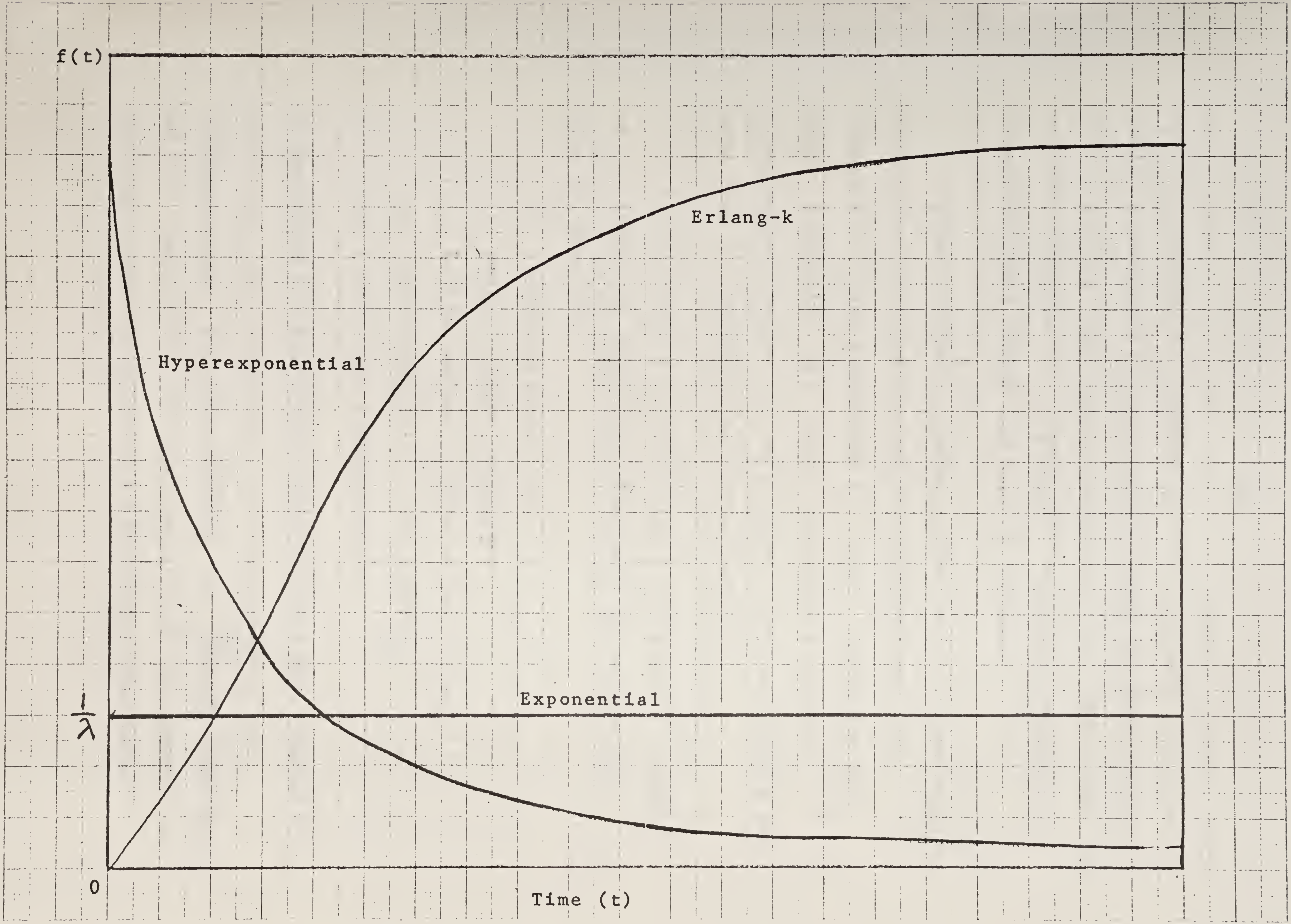


FIGURE 2. FAILURE RATES OF THEORETICAL DISTRIBUTIONS

multiprogramming and preemption were realized, only one job could be handled by the operating system at any one time. Other arriving jobs remained in queues until the current job had been processed to completion. Scheduling disciplines were relatively simple and easy to analyze.

Current systems, however, allow more than one ongoing task at any one time and the interruption and later resumption of the processing of a job. Scheduling has become difficult. The central processor as opposed to input/output devices, is still often a scarce resource but more than one processor may be in operation.

The difficulty of scheduling is illustrated somewhat by the number of objectives which must be kept in mind. Some of these are:

(1) minimum response time

(2) maximum system throughput

(3) reduced number of jobs in the system

(4) acknowledgement of job priorities

(5) low cost of implementation.

To determine how well these objectives are met, scheduling disciplines have been studied in one or more of the following ways: (1) analytically, (2) through simulation, and (3) empirically. In the remainder of this chapter we will categorize the scheduling disciplines into three classes and then in each discuss the research conducted employing these methods.

Types of Disciplines Formulated

For research purposes, scheduling rules have often been distinguished according to the amount and kind of information about a job needed to apply the scheduling rule. There are basically three classifications:

(1) those rules that require no information about the processing times of jobs,

(2) those rules that require exact information about the processing times of jobs,

(3) those rules that require knowledge of the probability distributions of the processing times of jobs.

The first class, i.e., those rules that require no information about a job, can be subdivided into classes that use no information gained about a job during its processing, and those that do. In the first group are FIFO, LIFO, PLIFO, and RAND. None of these discriminates among jobs on the basis of their length. In the second group are RR, PS, PPS, and FBn. These disciplines do discriminate among jobs by allowing short jobs to be completed before longer jobs.

The second class, i.e., those rules that require exact information about the processing times of jobs, can be subdivided into non-preemptive and preemptive disciplines. These include SPT and SRPT, respectively. Both process jobs in the order of their length, shortest to longest.

The third class, i.e., those rules that require knowledge of the probability distributions of the processing times of jobs, can also be subdivided into non-preemptive and preemptive disciplines. The first includes SEPT; the second includes SERPT and SR.

Analytical Studies

Assumptions and Performance Measures of Scheduling Models

Scheduling disciplines in each of the three groups have been studied under various constraints and with various performance measures in mind. Coffman (1967), Coffman & Kleinrock (1968), and McKinney (1969) have provided surveys of the models used and results obtained analytically, prior to 1969. McKinney (1969) has pointed out that scheduling models of time-sharing systems are stochastic and thus the models depend on queuing theory. Models may differ from each other in the following parameters:

(1) number of scarce resources, usually the processor,

(2) distribution of job arrivals, e.g., general or Poisson,

(3) handling of overhead (preemption),

(4) quantum assumption,

(5) processing time distribution, e.g., exponential or general, and degree of knowledge of this distribution,

(6) setup times,

(7) external priorities, and

(8) scheduling discipline.

Measures of performance obtained from these models include:

(1) average response time,

(2) average waiting time,

(3) average queue length,

(4) maximum waiting time, and

(5) weighted response times (linear losses).

Simple models have been found to portray accurately actual time-sharing system performance (Scherr 1967).

Optimality of Scheduling Disciplines

An important area has been the optimality of scheduling disciplines in the various models. This optimality may be examined from the viewpoint of minimizing total expected losses which is the same as maximizing the value of service rendered. We shall outline some of the optimality results that have been obtained through research, on each of the three classifications of scheduling disciplines discussed earlier.

For the class of scheduling disciplines with no initial information about the processing times of jobs, Schrage (1970) has found that:

(1) if no preemption is allowed,
    (a) FIFO is optimal for identical convex costs (i.e., every job has the same loss function which in this case is convex) and arbitrary, independent interarrival and processing times, and

(2) if preemption is allowed,
    (a) PPS is optimal for identical linear costs arbitrary, dependent interarrival times, and a decreasing failure rate,
    (b) FIFO is optimal for identical convex costs, arbitrary, dependent interarrival times, and an increasing failure rate,
    (c) PLIFO is optimal for identical concave costs, arbitrary, dependent interarrival times, and exponential processing times, and
    (d) RR is good but not necessarily optimal for identical convex costs, arbitrary, dependent interarrival times, and a decreasing failure rate.

For the class of scheduling disciplines with exact information on the service times of jobs, Smith (1956, cited in Sevcik 1971) and McNaughton (1959) among others have proven SPT/c, where c is a loss constant, optimal with preemption disallowed and no arrivals. With preemption allowed and identical linear loss functions, Schrage (1968) has proven SRPT optimal. SRPT/c is after the last arrival of a busy period when preemption is allowed. Under Poisson arrivals, SRPT/c is optimal in minimizing the expected rate of total loss (as opposed to total losses) (Sevcik 1971).

Finally, we consider the class of scheduling disciplines with knowledge of the service time distributions. Rothkopf (1966) and others have proven SEPT/c optimal, when preemption is disallowed and there are no arrivals. Sevcik (1971) has proven SR optimal in minimizing the expected rate of total loss under linear losses and Poisson arrivals, within a class of disciplines. It is conjectured that the optimal rule must lie within that class.

In the above cases, SR has been shown to specialize to the optimal scheduling discipline.

For more general models than these, scheduling problems become so complex that few optimality results have been obtained.

These three classes of scheduling disciplines have been investigated as to their performance under variables such as interarrival times, processing time distributions, utilization of the system, and loss functions. We will now present some of the important studies in this area.

Comparative Performance of Optimal or Good Scheduling Disciplines

Schrage (1969), Tran (1972), and Schrage & Miller (1966, cited in Schrage 1969) have investigated the effects of various conditions on scheduling disciplines. Schrage has considered a system with a single limited resource, the processor, exponential interarrival times, and identical linear loss functions. He has compared the following scheduling disciplines: FIFO, SPT, SRPT, RR, and FB with respect to the overall average response time and system load, and has found that FIFO, RR, and FB perform equivalently, but that the rules using information about processing times, SPT and SRPT, improve system performance greatly when a system is heavily loaded. In addition, he has investigated the effect of an increasing coefficient of variation and has found that full information (SPT/c versus FIFO) and preemption (RR versus FIFO and SRPT versus SPT/c) become more valuable. FIFO and FB are very distribution sensitive, with SPT/c, RR, and SRPT, less so.

Tran (1972) has simulated a system under the assumptions of a single processor, Poisson arrivals, linear loss functions, with and without overhead of preemption, and hyperexponential and bimodal normal processing time distributions. He compares three families of scheduling disciplines: (1) SPT, including preemption, (2) RR, including variations such as SQRR, CYCRR, and MQRR, and (3) FBn, with variations on the value n, with respect to mean response time, waiting time distribution, and response time distribution conditioned on service time. His results reveal that the SPT family is dominant as the scheduling strategy producing the "best" mean response times, waiting time distributions, and response time distributions both with and without overhead. They also indicate that response time distributions are significantly affected by service time distributions. Tran concludes that (1) the optimality of disciplines depends a great deal on the available information about service times (but not on the type of the particular service time distribution, i.e., here, hyperexponential versus bimodal normal) and (2) overhead costs do cause significant variation in the relative performance of scheduling disciplines.

Schrage and Miller have used information about processing times indicating which jobs are short and which are long. They apply external priorities favoring the short jobs and find that classifying 90% of all jobs as short, is responsible for 80% of the reduction in response times from FIFO to SPT. Schrage (1966, cited in Schrage 1969) has analyzed this situation when processing times are known only approximately and found that even this helps to reduce response times.

The above studies indicate that full information rules perform the best in attaining the objective of fast turnaround time. In the absence of full information about processing times, partial information in the form of service time distributions, is very beneficial.

Since exact information of processing times of jobs is seldom available in actual computer systems, in the remainder of this chapter, we will discuss research on empirical processing time distributions and the implications of these studies.

Service Time Distribution Studies

Analytical and simulation studies primarily assume that service time distributions have general exponential forms. Studies by Greenberger (1966), Scherr (1967), and Kleinrock (1967), among many others, have assumed exponential service time distributions. Tran (1972) has assumed hyperexponential. (He also considers a bimodal normal distribution.)

In actual empirical studies of service time distributions, general exponential forms do appear, but not the strictly exponential form assumed in so many models. Shemer & Heying (1969) have studied a batch time-sharing system of an SDS Sigma 5 and Sigma 7 to obtain empirical support for the validity of their proposed mathematical model. Without displaying any of the service time results, they assert that the empirical data does indicate that exponential service time distributions is a valid assumption for the model.

However, McIsaac (1965, cited in Schrage 1969), in an SDC study of an IBM 7094-type time-sharing computer has found that request time distributions could be approximated by a log-normal distribution. The coefficient of variation for the request times in this study, is large, viz., 17.45/1.45. Much variability exists among jobs and the distribution is skewed to the right.

Scherr (1966) has studied an MAC time-sharing system on an IBM 7094 and found that the distribution of request times has a mean of .88 and standard deviation of .7. The distribution has a greater probability near zero than has an exponential distribution with mean .88. No fitting of the empirical distribution was performed.

In a later SDC study, Fine, Jackson, & McIsaac (1966) have studied the number of instructions executed per request. They partition requests according to the specific purpose of the program, considering LISP, META5, GPDS, TINT, and SURF programs. While not actually studying the distribution of service times, they do study the variability of the number of instructions executed per request. Their results indicate that coefficients of variation range from .57 to 2.3. The statistics show distinct differences among the types of programs.

A JOSS study (Bryan 1967) of an interpretive system on a PDP-6 has revealed that the distribution of request times per session has a hyperexponential form with the mode at zero. The coefficient of variation was not given but the median was 7 seconds while the mean was 6.6 minutes on a typical distribution, showing much variability in processing time of requests.

Fuchs & Jackson (1970) have proposed an elementary model of the communication process between a multiaccess computer and a user at a terminal. They support this model with empirical data from four computer systems, investigating the distribution of user think times, computer burst times, interburst times, intercharacter times, and related discrete variables. Using parameters estimated from sample means and variances, they have found that all of the continuous variables can be represented by gamma distributions in all four systems, and some, namely, the user think time, computer burst time, and interburst time distributions, can be also represented by log-normal distributions. However their computer burst times included processing times plus time spent in queues and, in some cases, times between terminal "carriage returns" and "line feeds". No processing-time-only distributions were obtained.

Anderson (1972) has examined the service time distributions of an IBM APL interactive system on a 360/50 with the intent of combining these and other empirical results with theoretical research to improve system performance through an improved scheduling algorithm. He obtains an overall distribution of user input and then distributions of the input partitioned into three classes: compute transactions, commands, and output-bound transactions. His statistics show wide variation in all of the classes. The overall distribution has a coefficient of variation of 9.54, and the three partitions have coefficients of variation of 11.86, 6.57, and 3.23, respectively. All have long-tailed distributions. He also plots the mean residual service time distributions, the concentration curves, and the completion rates. The mean residual life is increasing for all input classes. This is followed by attempts to fit the service time distributions to hyperexponential and Weibull distributions, using cumulative proportions. No hyperexponential distribution is fit to the data since its parameters cannot be fit satisfactorily, by the method of moments. The Weibull distribution is used with parameters estimated by a regression method developed by Kac (1960, cited in Anderson 1972). No single Weibull distribution is fit but a composite of Weibull distributions does approximate the empirical distributions fairly closely, although not sufficiently well enough to satisfy a goodness-of-fit test.

Implications of Empirical Studies

Empirical service time distributions have resembled hyperexponential, gamma, Weibull, log-normal and Waring distributions. These distributions are similar in that their mean residual life times and, consequently, their ranks, are increasing throughout or increasing after some point in time. This can be attributed, in part, to the infinite tails of the distributions. Sevcik (1971) points out that it is possible to base service time distributions used in applying a scheduling discipline in a particular system on past empirical studies of that system. Therefore, we must consider the effects of these distributions on scheduling disciplines. When rank is decreasing, preemption is not necessary to obtain minimal expected loss. But when rank is increasing, preemption is desirable. Costs of preemption may become significant in the choice of a scheduling discipline, extremely so in the case of PS and PPS, less so in RR and FB, and still less so in SERPT.

One other observation of empirical studies is of major importance in considering the applicability of SR or any other distribution-dependent discipline. In most of the studies, much variability in the processing times of jobs has appeared, evident in the large coefficients of variation and the very long tails of the distributions. Consequently, moments of the distributions are quite inaccurate. Occurences in the tail can significantly alter the mean, standard deviation, skewness, and kurtosis. Other statistics used to describe distributions, such as failure rate and mean residual life are also affected. As mentioned above, mean residual life may be increasing when it is desirable to have it decreasing. Sevcik (1971) conjectures that by partitioning arriving jobs into classes with common a priori characteristics, distributions having lower coefficients of variation than the overall distribution may be obtained. Anderson's work (1972) discussed earlier, does indicate this effect, although his research partitioned the overall distribution only on the basis of the type of time-sharing command. The coefficient of variation of the main distribution was reduced in two of the three partitioned classes. In the event of the coefficient of variation not being reduced significantly, if distinct distributions are obtained in the partitioned classes, more accurate information is known about the processing time distribution of the jobs within each class. To determine if information known at the arrival of a job about that job--information such as priority, amount of core required, type of user--affects the distribution of the processing times of the jobs, directed research is necessary.

In the remainder of this thesis, this partitioning concept will be investigated in a batch processing system at the University of Toronto. Chapter II presents the method used to gather and process data and the parameters considered in partitioning the data. Measurements used to determine differences in distributions are outlined.

Chapter III reviews the experiments conducted on the data gathered and the results obtained from these experiments. Distinctive or unusual observations are mentioned.

Chapter IV outlines the conclusions reached based on the results of Chapter III.

CHAPTER II

Objectives of Our Investigation

By collecting a large amount of information concerning jobs in an actual system, we hoped to obtain the service time distribution of these jobs, and then condition this distribution on one or pairs of characteristics of the jobs. Our objective was to determine which, if any, of the parameters would be indicative of the service time distribution of a job. To determine scheduling performance implications, we tried to characterize the empirical distributions by attempting to fit them to theoretical distributions and to calculate their associated functions (i.e., mean residual life and smallest rank).

Methods Employed in the Processing of the Data

The nature of our problem requires the processing of large amounts of data to obtain frequency distributions and associated statistics and then the comparison of these to known probability distributions. Various methods are employed in the areas of empirical distribution fitting and measurement of differences in empirical distributions. Before discussing the actual data handling procedures, we give descriptions of these methods below.

Distribution Fitting

A. Smoothing

Before empirical distributions can be tested against theoretical distributions, extreme fluctuations in the frequency classes must be eliminated as they are isolated occurrences not indicative of the actual population distribution. These fluctuations are often referred to as "noise". To eliminate "noise", a numerical method known as least square polynomial approximation can be used. Hildebrand (1956) and Ralston (1965) discuss the following:

(1) Three point linear approximation

(2) Five point linear approximation

(3) Approximation by a polynomial of degree greater than one.

A brief discussion of each is given below.

(1) Three point linear approximation. Given function values $f_1, f_2, \ldots, f_n$ at equidistant points $x_1, x_2, \ldots, x_n$, then the smoothed values $y_1, y_2, \ldots, y_n$ can be calculated from the formulas

$$y_i = 1/3(f_{i-1} + f_i + f_{i+1}), \quad i = 2, \ldots, n-1,$$

$$y_1 = 1/6(5f_1 + 2f_2 - f_3),$$

$$y_n = 1/6(-f_{n-2} + 2f_{n-1} + 5f_n).$$

This method may be adjusted to non-equidistant points, $x_1, \ldots, x_n$. It is used where linearity over three points is a good assumption. Although a function may not be linear, this method applied over subranges of the data would smooth the function satisfactorily. It may also be applied repeatedly, obtaining greater smoothing and eventually a straight line.

(2) Five point linear approximation. For formulas, see Hildebrand (1956). This method in application is similar to the three point linear approximation. However, its effect on smoothing is more pronounced. It should be applied only when linearity over five points is a good assumption.

(3) Approximation by a polynomial of degree greater than one. This method is used when the true function is thought to be a non-linear polynomial. The polynomial may be chosen so that it fits the observed data exactly, but this would reflect "noise". Thus a polynomial of degree less than the number of function values should be chosen, based on intuitive knowledge of the true function.

B. Parameter Estimation

Empirical distributions can generally be fit to theoretical distributions using the method of moments or the method of maximum likelihood to determine the parameters of the theoretical distribution.

The method of moments is the oldest and simplest method. For a k parameter distribution, it consists of determining the first k moments of the theoretical distribution which are generally functions of the parameters, and the first k moments of the empirical distribution. Then by equating each of the k moments, k simultaneous equations in k unknowns are obtained which can be solved for the k parameters.

The method of maximum likelihood depends on obtaining a value of the unknown parameter, θ, which maximizes a likelihood function, L, defined as:

$$L = P(X_1=x_1)P(X_2=x_2)\ldots P(X_n=x_n) \quad \text{where}$$

$X_1, X_2, \ldots, X_n$ is a random sample from a discrete probability distribution and the X_i's are the actual values of the random variables, X_i. For a continuous distribution, $P(X_i=x_i)$ is replaced by the density function of the random variable X_i at the sample point x_i. This method can be said to obtain the value of θ which maximizes the probability of obtaining the particular sample. For further discussion of this method and its advantages, see Hillier and Lieberman (1967).

Parameter estimation methods pertaining to a specific distribution are often used. One to estimate the parameters of a Weibull distribution has been developed by Kao (1960, cited in Anderson 1972), based on the following property of the Weibull distribution:

$$W = \ln \ln(1/(1-F(t))) = A \ln t - A \ln B$$

where A and B are the parameters of the distribution. By letting $z = \ln t$, $x = A \ln B$ and $y = A$, and by performing a regression of $W = yz - x$ on z, estimates of x and y are obtained. Then solving $A = y$ and $B = \exp(-x/y)$ for A and B yields the estimates of the parameters A and B.

D. Goodness-of-Fit Tests

To determine how well an empirical distribution is approximated by a theoretical distribution, a goodness-of-fit test can be performed. One such test commonly used is the chi-square test. It is based on the calculation of the differences of the empirical distribution values from the theoretical distribution values. For large samples, the following

$\sum ((f_o - f_e)/f_e)$ where f_o is the empirical

distribution value and f_e is the theoretical value, for large samples, has a chi-squared distribution with k-1 degrees of freedom where k is the number of terms in the sum. If it is necessary to calculate the parameters of the theoretical distribution from the empirical distribution, then the best possible estimates of the parameters, used in the same manner as actual parameter values would be, will result in a chi-square distribution with k-1-p degrees of freedom where p is the number of estimated parameters.

Measurements of Differences in Empirical Distributions

To detect differences in empirical distributions, the means, standard deviations, and coefficients of variation (standard deviation/mean) can be compared. If the distributions do not have very long tails, skewness and kurtosis may also be considered. For more intuitive and complete information, the actual shapes of the distributions as illustrated by the cumulative proportions and graphical representations, can be examined. In this regard, the height of the peak, the breadth of the peak, the proportion of jobs lying in the peak, the maximum point of the peak, and the fullness and length of the tail are important.

Data Gathering and Handling

Source of the Data

The data for this research was obtained from the University of Toronto accounting records over a thirty day period in November and December of 1972. The accounting records contain information from the HASP/OS-MVT system operating on an IBM 370/165. HASP manages jobs from three sources. A brief description of each follows.

(1) General purpose job stream (GPJS). This is a batch processing facility that handles jobs requiring utilities and language processors such as PL/I, COBOL, FORTRAN, RPG, and Assembler. A job is submitted with a job card followed by job control language (JCL) cards, providing information required to process the job. Much of this information and additional information gathered during the processing of the job, is stored in the accounting records. These jobs require an amount of computer time limited by user class specifications.

(2) High speed job stream (HSJS). This is a batch processing facility that handles short jobs requiring special high-speed langauge processors such as WATFIV, PLUTO, ALGOLW, and PL/C. It provides very fast turnaround service free of charge. A job card providing very limited information is submitted with a job. This information plus limited information gathered during the processing of the job is stored in the accounting records.

(3) Conversational remote job entry (CRJE). This is a system which interfaces with HASP to allow jobs from remote stations linked to the IBM 370 via low-speed communication lines to be submitted for batch processing by HASP.

Description of the Data Gathered

Accounting information is stored on a daily basis, i.e., one 24-hour period is replaced by the next 24-hour period. Thus, when a 24-hour period had passed, a PL/I program was submitted, which examined the information from this time period, extracted a number of the accounting items of interest in this research, and stored these on a tape. This was repeated until over 30,000 jobs from the general purpose job stream had accumulated.

For each job in the general purpose job stream, the following information was recorded:

(1) priority. A user submitting a job to the system requests a priority of 4, 6, or 8 for that job. A priority of 6 is charged at a normal rate; a priority of 8 is twice the charge for a priority of 6; and a priority of 4 is .6 times the charge for a priority of 6. Priority 8 provides the fastest turnaround time.

(2) source. Various remote terminals exist at the University and in surrounding areas from which HASP jobs are submitted. These jobs are assigned a two letter code corresponding to the various terminals.

(3) authorization code. Every job has an account authorization letter from one of the following areas: (a) administration, (b) commercial, (c) undergraduate course work, (d) graduate course work, (e) research, (f) other universities, (g) other than university or commercial but having some affiliation with universities, and (h) Computing Centre staff work.

(4) authorization code number. Every job is assigned an account number along with the account letter. These numbers range from 1 to 2000.

(5) number of cards read in.

(6) estimated execution time for the job. This includes CPU time plus the time spent waiting for I/O to be accomplished.

(7) estimated number of lines to be printed.

(8) estimated number of cards to be punched.

(9) amount charged for the job as a function of core allocation, CPU time and I/O time. If CPU time and I/O time are not both zero, core allocation is obtained from the amount charged for the job.

(10) the time at which a job was passed to the operating system.

(11) the class of a job. A job class can be A, B, C, or M. A class A job has constraints of:

(1) ≤ 200K of core (this is not enforced),

(2) ≤ 5 minutes total run time,

(3) no tape and/or disk mounts,

(4) ≤ 15,000 printed lines, and

(5) ≤ 2000 cards punched.

A class B job has constraints of:

(1) ≤ 200K of core,

(2) ≤ 5 minutes total run time,

(3) 1 7-track tape and/or 9-track tape and/or one disk mount,

(4) ≤ 15,000 printed lines, and

(5) ≤ 2000 cards punched.

A class C job has no constraints. The job class M is used primarily by Computing Centre staff when immediate turnaround is imperative, usually in cases dealing with the repair or maintenance of the operating system.

Program Description

A PL/I program was developed to process this data. The objective of the program is to obtain distributions of the processing times (i.e., CPU times, and hereafter referred to as CPUT) of jobs, and since we are dealing with a multiprogramming environment, distributions of the total run times (i.e., CPU time plus I/O time, and hereafter referred to as TRT) of the jobs. These distributions are conditioned on zero, one, or two of the accounting information items outlined previously.

The program consists of the following sections or procedures:

(1) Determination of the CPUT interval or TRT interval. Only general purpose job stream jobs are considered. High speed job stream jobs are eliminated because CPU times and I/O times are not recorded.

(2) Determination of the conditioning parameter class.

(3) Tabulation of the appropriate frequency matrix class (determined by (1) and (2) above). These classes are in the form of buckets. A bucket depends upon the number of conditioning parameters and the nature of the parameter, i.e., discrete or continuous. By discrete we are referring to parameters whose values cannot be grouped through interval classification, e.g., priority and class. By continuous, we are referring to parameters whose values can be grouped through interval classification. A bucket can represent:

(a) a CPUT or TRT interval,

(b) a CPUT or TRT interval and one (or two) discrete parameter value(s),

(c) a CPUT OR TRT interval and one (or two) parameter interval(s), or

(d) a CPUT or TRT interval, one discrete parameter value, and one parameter interval.

(4) Smoothing of the frequency distributions. A three-point linear approximation technique applied twice in succession, is used. This is considered a good method because in areas of distribution peaks, the function values are spread over small intervals, and at larger intervals, i.e., the tails of the distributions, the shape of the distributions is generally linear.

(5) Distribution statistics. These include the mean, standard deviation, skewness, and coefficient of variation, calculated using grouped data methods.

(6) Proportion distribution. The proportion in each frequency interval and the cumulative proportion up to and including that interval, are calculated.

(7) Distribution characterization. The mean residual life and rank of each distribution are calculated.

(8) Chi-square test.

(9) Theoretical distribution calculations. These deal with the exponential, hyperexponential, and gamma distributions. The parameter estimates are obtained as

follows:

(a) exponential: $1/\lambda$ = mean of empirical distribution. This is a maximum likelihood estimator.

(b) hyperexponential: the method of moments is used to calculate λ_1 and λ_2. The estimates are:

$$\lambda_1 = (4pm \pm 16p^2m^2 - 8p(2m^2 - (1-p)(sd^2 + m^2))/$$

$$(4m^2 - 2(1-p)(sd^2 + m^2))$$

and $$\lambda_2 = (\lambda_1(1-p))/(\lambda_1 m-p)$$

where m and sd are the mean and standard deviation, respectively, of the empirical distribution. p assumes values from 0 to 1.

(c) gamma: the method of moments is used to obtain

$$A = m^2/ sd^2 \quad \text{and} \quad B = sd^2/m.$$

Matching of the mean and maximum rather than the mean and variance produces

$$A = m/(m-max) \quad \text{and} \quad B = m-max$$

where max is the point at which the empirical distribution attains its maximum. These are also considered in the program.

Some Implementation Considerations of the Program

The data from the accounting tape is in bit, character or fixed binary form. To process this information in an efficient manner, overlay storage techniques are used which enable comparison on a bit basis rather than a byte or word basis.

The frequency distribution is in the form of an m by n matrix where m is the CPUT or TRT interval and n indicates a conditioning parameter class. If there is only one conditioning parameter, then the columns range from 1 to the number of classes for that parameter. If there are two parameters, then the columns range from 1 to pq where p is the number of classes for the first parameter and q is the number of classes for the second parameter. Other calculations are in table form.

Program Output

A typical run would produce the following:

```
FREQUENCY FOR CLASS 3 IS 2508
MAXIMUM IS 15.50
MEAN FOR CLASS 3 IS 409.72
STANDARD DEVIATION FOR CLASS 3 IS 3543.90
SKEWNESS FOR CLASS 3 IS 19.29
COEFFICIENT OF VARIATION IS 8.6495

CPU PARM FREQ SMED  %   CUM       MRL      RANK     EXP   CLASS
CL   CL         FREQ    PROB                        DIST   MARK

  1   3     0     0  0.0 0.00     409.7      52.7     .0      0.0
  2   3   114   114  4.5 0.05     423.5      39.8   63.5      5.5
  3   3   602   602 23.9 0.29     552.3     102.9   58.9     15.5
  4   3   199   174  6.9 0.35     600.7     139.8   57.5     25.5
  5   3    66   116  4.6 0.40     636.2     210.6   56.1     35.5
  6   3    65    71  2.8 0.43     657.4     322.8   54.8     45.5
  7   3    50    44  1.8 0.45     668.1     407.4   53.5     55.5
  8   3    16    31  1.2 0.46     673.1     406.4   52.2     65.5
  9   3    21    23  0.9 0.47     674.8     403.2   50.9     75.5
 10   3    23    22  0.9 0.48     676.0     399.6   49.7     85.5
 11   3    20    24  1.0 0.49     678.4     396.7   48.5     95.5
 12   3    36    25  1.0 0.50     676.8     388.9   93.5    110.5
 13   3    21    24  0.9 0.51     669.3     375.0   89.1    130.5
 14   3    18    22  0.9 0.52     660.9     360.5   84.8    150.5
 15   3    21    20  0.8 0.52     651.5     345.1   80.8    170.5
 16   3    23    18  0.7 0.53     641.3     329.0   76.9    190.5
 17   3    11    15  0.6 0.54     626.8     309.1   91.0    213.0
 18   3    11    14  0.6 0.54     609.0     285.9   85.6    238.0
 19   3    10    18  0.7 0.55     593.4     263.6   80.6    263.0
 20   3    26    28  1.1 0.56     582.9     242.9   75.8    288.0
 21   3    59    55  2.2 0.58     574.0     211.7  138.4    325.5
 22   3    60   122  4.9 0.63     593.4     174.1  122.5    375.5
 23   3   180   202  8.1 0.71     695.4     143.4  108.4    425.5
 24   3   423   252 10.1 0.81     990.8     159.2   95.9    475.5
 25   3   266   222  8.8 0.90    1731.3     203.4  160.1    550.5
 26   3    17   122  4.9 0.95    3209.0     264.7  125.5    650.5
 27   3     9    48  1.9 0.97    4996.6     756.1   98.3    750.5
 28   3    21    16  0.6 0.97    6045.9    1113.6  137.3    900.5
 29   3    20    14  0.6 0.98    7473.8    1356.5  113.3   1150.5
 30   3     5    11  0.4 0.98    9210.9    1560.6   54.5   1450.5
 31   3     6     7  0.3 0.99   11031.6    1615.6   26.2   1750.5
 32   3     8     6  0.2 0.99   13113.0    1573.4   12.6   2050.5
 33   3     2     5  0.2 0.99   15831.6    1905.2    6.6   2350.5
 34   3     5     4  0.2 0.99   19532.6    3790.4    3.9   2750.5
 35   3     5     3  0.1 0.99   23415.8    6560.5    1.5   3500.5
 36   3     0     2  0.1 0.99   26447.1    9349.5    0.1   4500.5
 37   3     1     1  0.0 1.00   28494.9    9784.4    0.0   5500.5
 38   3     0     1  0.0 1.00   30193.7    8961.5    0.0   6500.5
 39   3     3     1  0.1 1.00   34462.0   10233.9    0.0   8000.5
 40   3     1     2  0.1 1.00   42293.9   11320.6    0.0  10500.5
 41   3     3     2  0.1 1.00   62729.1   36029.1    0.0  15000.5
```

```
42     3     3     2  0.1 1.00 116000.0 116000.0    0.0  39000.5
43     1     1     1  0.0 1.00                        0.0 155000.5

CHISQUARE VALUE WITH 33 D.O.F. IS 65.86
```

The above table is the CPUT distribution for a Core Allocation class of 40-60K. The SMED FREQ column presents the values resulting from smoothing the distribution represented in the FREQ column. Results are in hundredths of seconds. Because such tables are difficult to read, we represent the results in the form of graphs which depict the variations in the distributions.

CHAPTER III

Experiments and Results

Various experiments were conducted on the data and the resulting distributions. They are discussed under four general topics:

(1) representative distributions,

(2) best partitions,

(3) distribution fitting, and

(4) distribution characterizations.

Results are presented with the experiments. A section of general observations based on the results in (1) is presented under that topic.

Representative Distributions

By representative distributions we mean distributions that meaningfully reflect the effects of conditioning CPUT or TRT on zero, one, or two parameters. To obtain these representative distributions, we first conducted experiments in two areas:

(1) interval adjustment involving CPUT and TRT, and

(2) interval adjustment involving continuous parameter classes.

These are discussed below.

(1) Interval Adjustment Involving CPUT and TRT

Our objective in adjusting these intervals was to obtain maximum information of the distribution of the jobs, e.g., the peak(s) of the distribution, and yet to maintain a manageable number of classes. Our method involved first obtaining the upper limits of CPUT and TRT and then selecting initial intervals. An interval for times of zero seconds was included to isolate jobs that are terminated before requiring any processor time (due, most likely, to a JCL error). The number of intervals for the CPUT and TRT were, initially, 17 and 18, respectively. This was increased gradually to obtain a more equal distribution over all the intervals, through increasing the number of intervals in areas of great concentration of jobs.

A final set of 43 intervals for both CPUT and TRT emerged. This handles reasonably well, all significant distribution classes (frequency greater than 50) resulting from unconditioned or conditioned CPUT and TRT, fairly well. Those classes with an extremely high number of observations are well distributed over the intervals. Those classes with an extremely low number of observations (50 to 100) are distributed relatively evenly, although some intervals may not contain any jobs. Classes of frequency less than 50 are not considered to be conclusive distributions of that area of a total population. Any significantly greater number of intervals is felt to hamper visualization of the general shape of the distributions.

The decision to have only one set of intervals for both CPUT and TRT was made primarily to aid comparison of the distributions for each of these variables. This has little effect on the actual distributions or relevant statistics. Those intervals of low CPUT with high incidences of jobs, contain fewer jobs in TRT distributions. Those intervals with high incidences of jobs in TRT distributions, contain fewer jobs in CPUT distributions.

(2) Interval Adjustment Involving Continuous Parameter Classes

Our objectives in adjusting these intervals were:

(a) to locate any differences in the CPUT and TRT distributions conditioned on these intervals,

(b) to obtain relatively even total frequencies in each class, and

(c) again to maintain a manageable number of classes.

To illustrate the problem, consider CPUT distributions conditioned on the amount of core allocated. A core allocation class of 100-120K produces a hyperexponential-like distribution. However, if the class interval is increased to 100-150K, the distribution becomes somewhat gamma-shaped where the mode is removed from the first interval. And if the class interval is decreased to 100-110K, then a small total class frequency indicates an unknown distribution.

Our method was similar to that in section (1). Estimates of the average value and the upper bound of the parameters were initially available. With these two guidelines in mind, the initial set of intervals was chosen. After a number of runs using these intervals, they were modified in line with our objectives. Table I below gives the upper bound of each interval in the set of intervals for each of the continuous parameters considered.

TABLE I

# of Cards Read In	Estimated Execution Time (sec.)	Estimated # of Lines Output	Estimated # of Cards Punched	Amount of Core Allocated (K)	Time of Execution
50	10	500	5	10	2 A.M.
100	20	1000	50	20	4
150	30	2000	100	40	6
200	40	3000	150	60	8
250	50	4000	200	80	9
300	60	5000	300	100	10
800	80	6000	400	150	11
1000	100	8000	600	200	12
4000	150	10000	800	300	1 P.M.
60000	240	15000	1000	500	2
	420	60000	1200	1100	3
	600	500000	1500		4
	900	1000000	2000		5
	3000		3000		6
	43200		60000		7
	86400				8
					9
					10
					11
					12
----	----	----	----	----	----
10	16	13	15	11	20

Number of Classes

Results

Before discussing the distributions, one relevant observation is of importance. The bucket method used to obtain our distributions is not felt to affect our results significantly. As we have employed it, in areas of a high concentration of jobs, a larger number of buckets have been used, as opposed to the number used in low concentration areas. For example, from 0 to 1 seconds, we use 10 buckets; from 10 to 100 seconds, we use 11 buckets. In this manner, we can obtain a fairly accurate idea of the shape of the distribution.

The distributions obtained from conditioning CPUT and TRT on the continuous parameters and also on the discrete parameters were examined to determine which parameters would provide good information about the processing time distribution of jobs before their actual processing. The criteria of a good parameter are:

(1) Distinct processing time distributions.

(2) If discrete, the number of classes is less than 25.

(3) No single default is taken by most of the jobs.

The following parameters were eliminated:

(1) source (greater than 25 classes with low frequency in some).

(2) authorization number (greater than 1000 classes with low or zero frequency in some classes).

(3) time at which a job was passed to the operating system (distributions are similar for all intervals except 10-11 P.M. and 11-12 P.M. which are times used by the Computing Centre staff)..

(4) estimated number of cards punched (less than 20%

of the jobs had estimates other than default 400 cards).

The parameters remaining for further investigation were:

(1) priority,
(2) authorization code letter,
(3) number of cards read in,
(4) estimated number of lines output,
(5) estimated execution time,
(6) amount of core allocated, and
(7) class.

Best Partitions

Conditioning on One Parameter

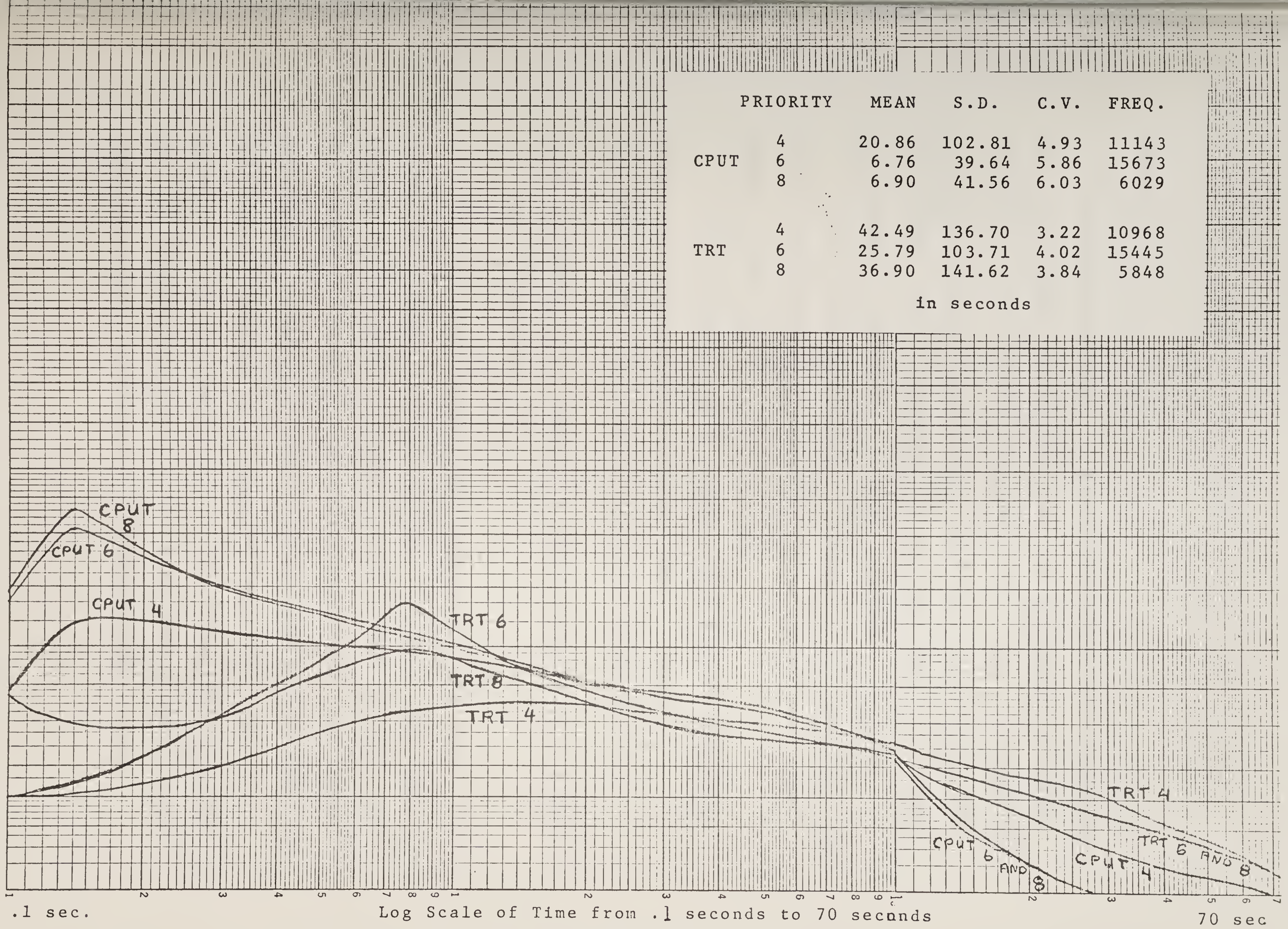
The seven parameters considered as good indicators of the processing time distributions of jobs were next examined from the viewpoint of partitioning the representative distributions into distinctly different distributions. This involved measuring the differences in the distributions to determine which, if any, should be combined.

Of the discrete parameters, i.e., Priority, Authorization Code Letter (hereafter referred to as Authorization Code), and Class, Priority was eliminated from further investigation and Authorization Code was adjusted.

Priority was dropped because of its 3 classes, only two distributions can be considered as distinct from each other. Priority 4 with 11,000 jobs, emerges as one distribution, while Priorities 6 and 8 with the remaining 22,000 jobs emerge as the other. Figure 3 shows the three CPUT and TRT distributions. (Note the remarks pertaining to this figure under Results below). Other parameters, because of their greater number of partitions, were considered more informative as to the processing time distributions of jobs entering the system.

Authorization Code produces relatively distinct distributions of sufficient frequency in 6 of its 8 classes, i.e., in commercial, administrative, graduate, undergraduate, research, and Computing Centre staff. But the classes, other-university and university-associated, have less than .3% of the jobs in each. Since other-university is similar to graduate, and university-associated is similar to Computing Centre staff, these classes were combined.

Of the continuous parameters, i.e., Number of Cards Read In, Estimated Number of Lines Output, Estimated Execution Time, and Core Allocation, Estimated Execution Time and Core Allocation were adjusted. The number of Estimated Execution Time classes was reduced from 16 classes to 8. The number of classes for Core Allocation was reduced



	PRIORITY	MEAN	S.D.	C.V.	FREQ.
	4	20.86	102.81	4.93	11143
CPUT	6	6.76	39.64	5.86	15673
	8	6.90	41.56	6.03	6029
	4	42.49	136.70	3.22	10968
TRT	6	25.79	103.71	4.02	15445
	8	36.90	141.62	3.84	5848

in seconds

FIGURE 3. CPUT AND TRT CONDITIONED ON PRIORITY

from 11 to 8.

The distributions of Estimated Number of Lines Output are all distinct to some degree, but intervals 500-1000, 2000-3000, and 4000-5000 resemble each other somewhat, and likewise, intervals 1000-2000, 3000-4000, 6000-8000, and 8000-10000. This parameter was not considered in further conditioning experiments for the following reasons:

(1) Its large number of classes (13) means that with conditioning on two parameters, up to 130 classes could result, with very few jobs in many.

(2) A relationship between CPUT and the Estimated Number of Lines Output is not evident. Instead, as mentioned above, the distributions are quite erratic.

Other parameters are better in these respects, as is evident in later discussions.

The distributions of the Number of Cards Read In prove to be all distinct to some degree. However, here again the definite relationship existing between CPUT and both Core Allocation and Estimated Execution Time is not as evident. The greater number of classes (10) as compared to the 8 of both Core Allocation and Estimated Execution Time is a deterrent from considerations in two parameter conditioning. The ten distributions tend to be much more similar to each other than those of Estimated Execution Time or Core

Allocation. Since it does not perform as well as the other continuous parameters, it was not considered for further investigation.

Results

The final set of partitions for each of the parameters is given below. Any job in a particular interval will have a value greater than the lower bound and less than or equal to the upper bound.

TABLE II

# of Cards Read In	Estimated # of Lines Output	Estimated Execution Time (sec.)	Amount of Core Allocated (K)
0-50	0-500	0-10	0-10
50-100	500-1000	10-20	10-40
100-150	1000-2000	20-60	40-60
150-200	2000-3000	60-150	60-80
200-250	3000-4000	150-240	80-130
250-300	4000-5000	240-420	130-200
300-800	5000-6000	420-900	200-300
800-1000	6000-8000	900-43200	300-1100
1000-4000	8000-10000		
4000-60000	10000-15000		
	15000-60000		
	60000-500000		
	500000-1000000		
10	13	8	8

Number of Classes

Priority: 4, Combined 6 and 8

Class: A, B, C, M

Authorization Code: Administration
Commercial
Undergraduate
Research
Graduate
Computing Centre Staff

Figures 6 to 14 depict the density function distributions of parameters considered in further experiments, based on these partitions. In examining these figures, the following should be kept in mind:

(1) The paper used is probability (ordinate) by 2 log (abscissa) cycles. The log abscissa is necessary to portray distributions relevant over a long interval of time, viz., 0

to 2500 seconds, and concentrated in a small portion of that interval, viz., 0 to 2 seconds, primarily. The probability ordinate is essential in distinguishing variations in the low proportions of jobs in the distribution tails.

(2) Using this paper implies that the zero value of the distributions is not represented. The ordinate begins with a .0001 probability, and the abscissa with .1 seconds.

Figures 4 and 5 give examples of distributions drawn to a normal scale. The long tails are not represented because of graph size considerations.

Conditioning on Two Parameters

Conditioning on combinations of two of the following four parameters was performed to determine if the resulting distributions were again distinct:

(1) Class

(2) Authorization Code

(3) Amount of Core Allocated

(4) Estimated Execution Time.

Results

The resulting distributions are shown in Figures 15 to 26. These "graphs" are intended to portray the general

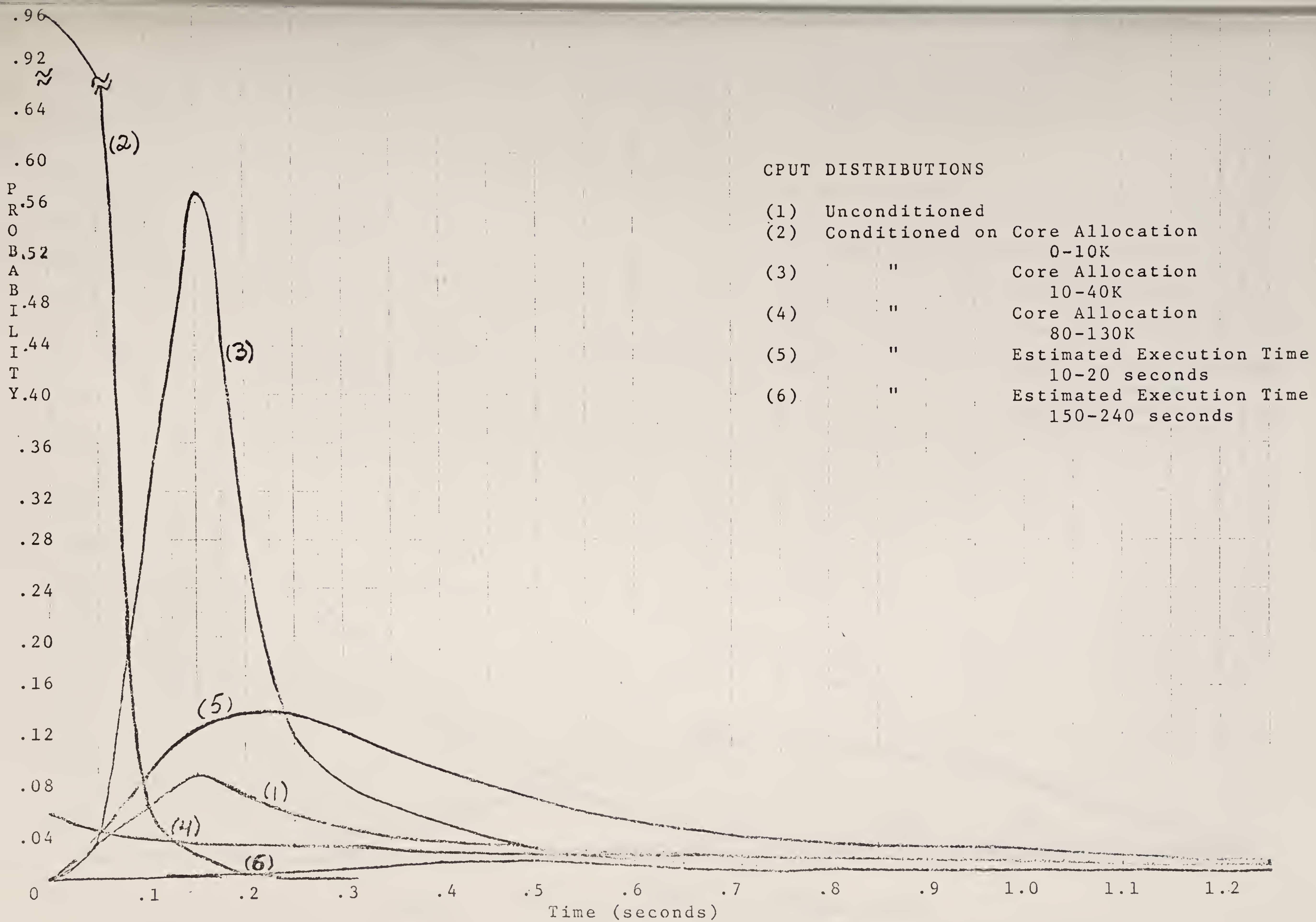


FIGURE 4. CPUT DISTRIBUTIONS ON A NORMAL SCALE

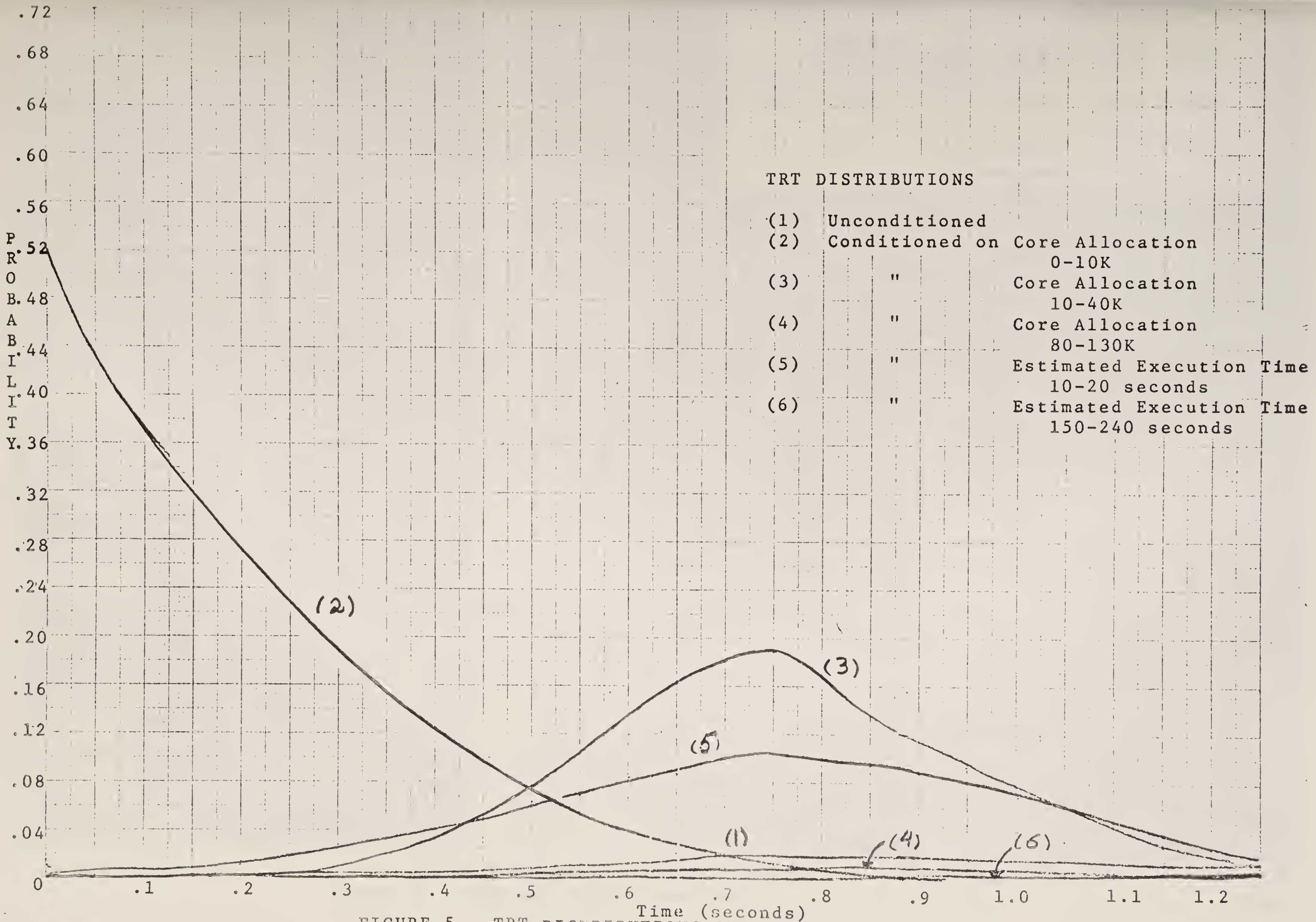
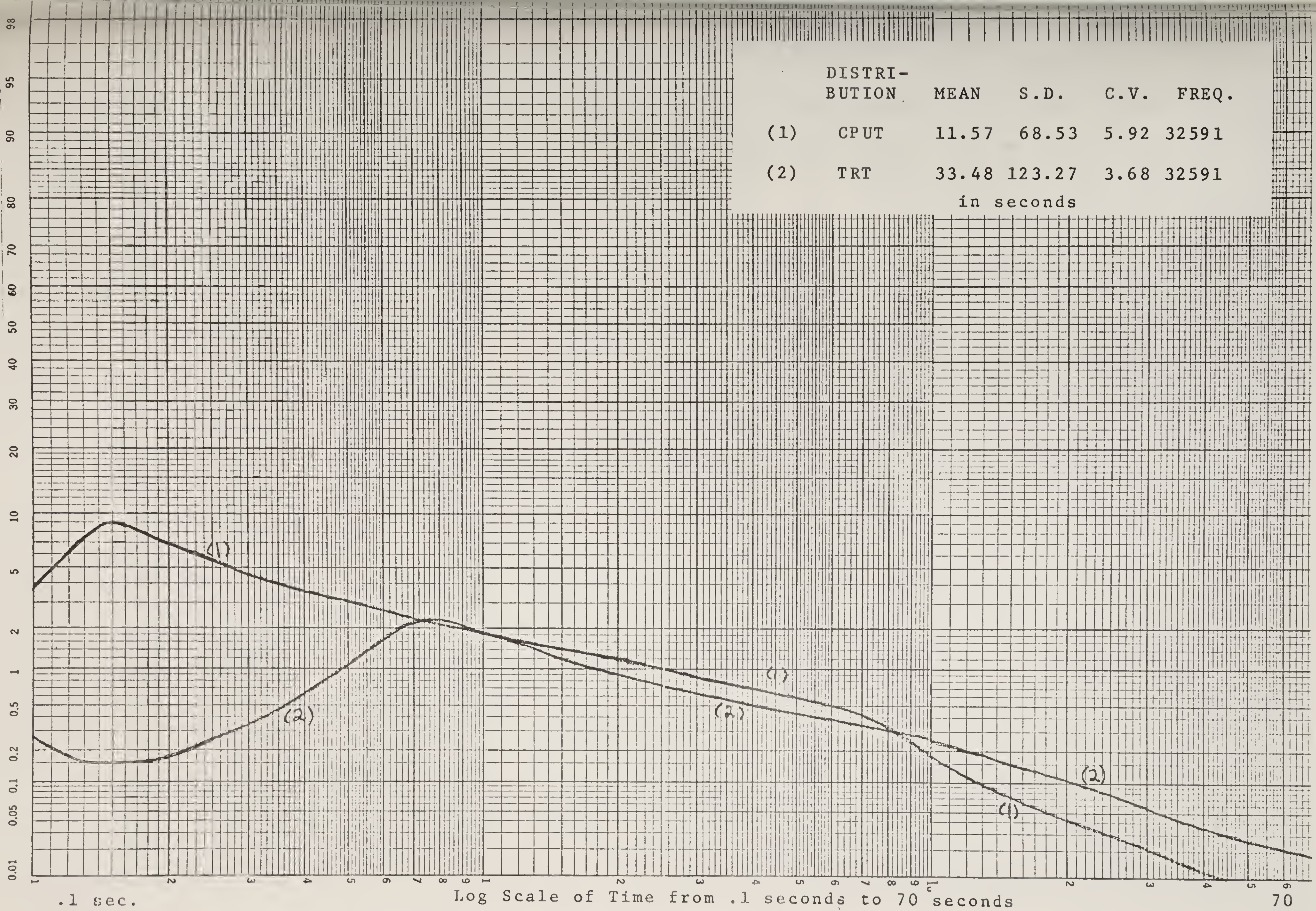


FIGURE 5. TRT DISTRIBUTIONS ON A NORMAL SCALE

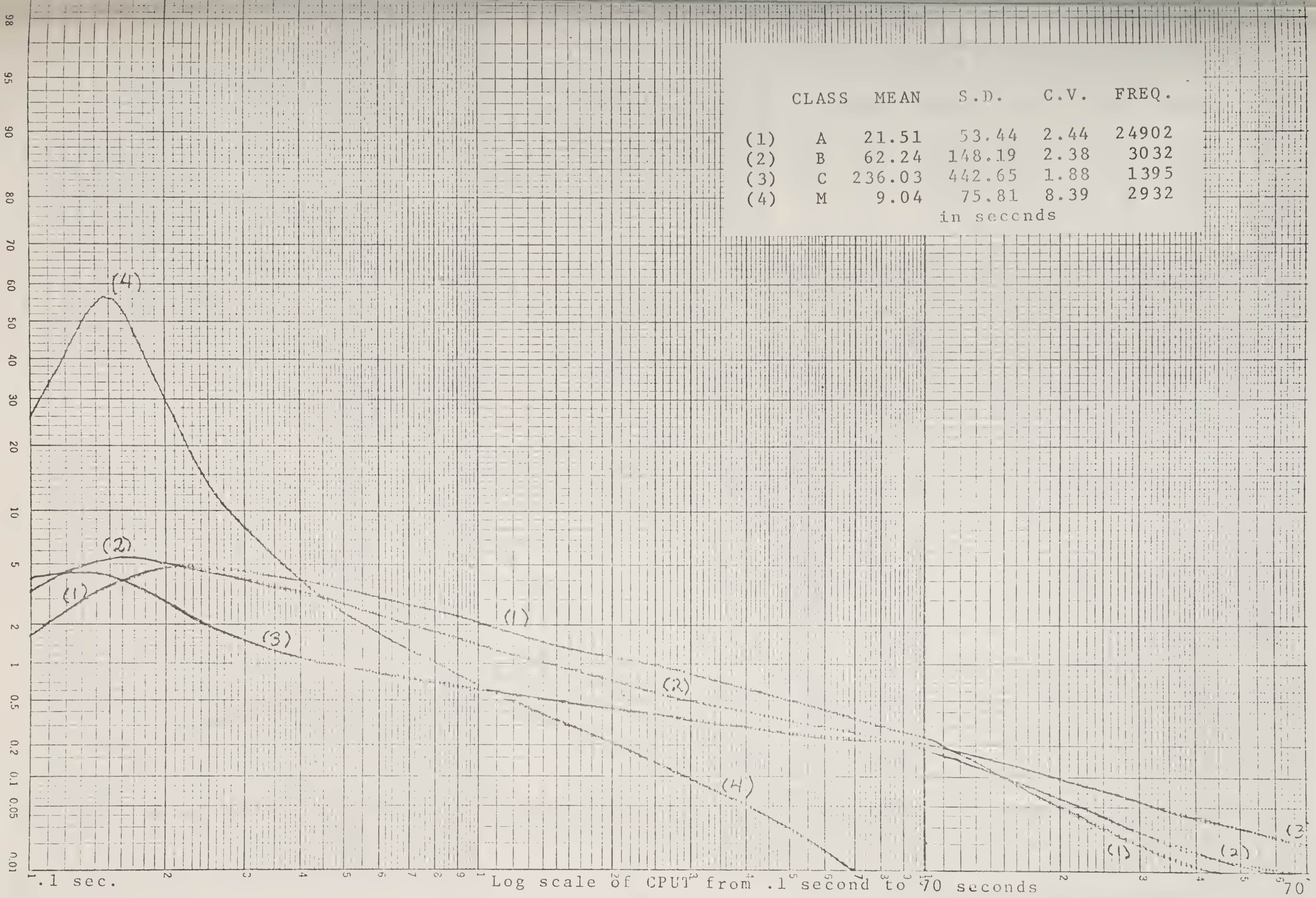


DISTRI-BUTION		MEAN	S.D.	C.V.	FREQ.
(1)	CPUT	11.57	68.53	5.92	32591
(2)	TRT	33.48	123.27	3.68	32591

in seconds

FIGURE 6. CPUT AND TRT UNCONDITIONED

Probability x Log

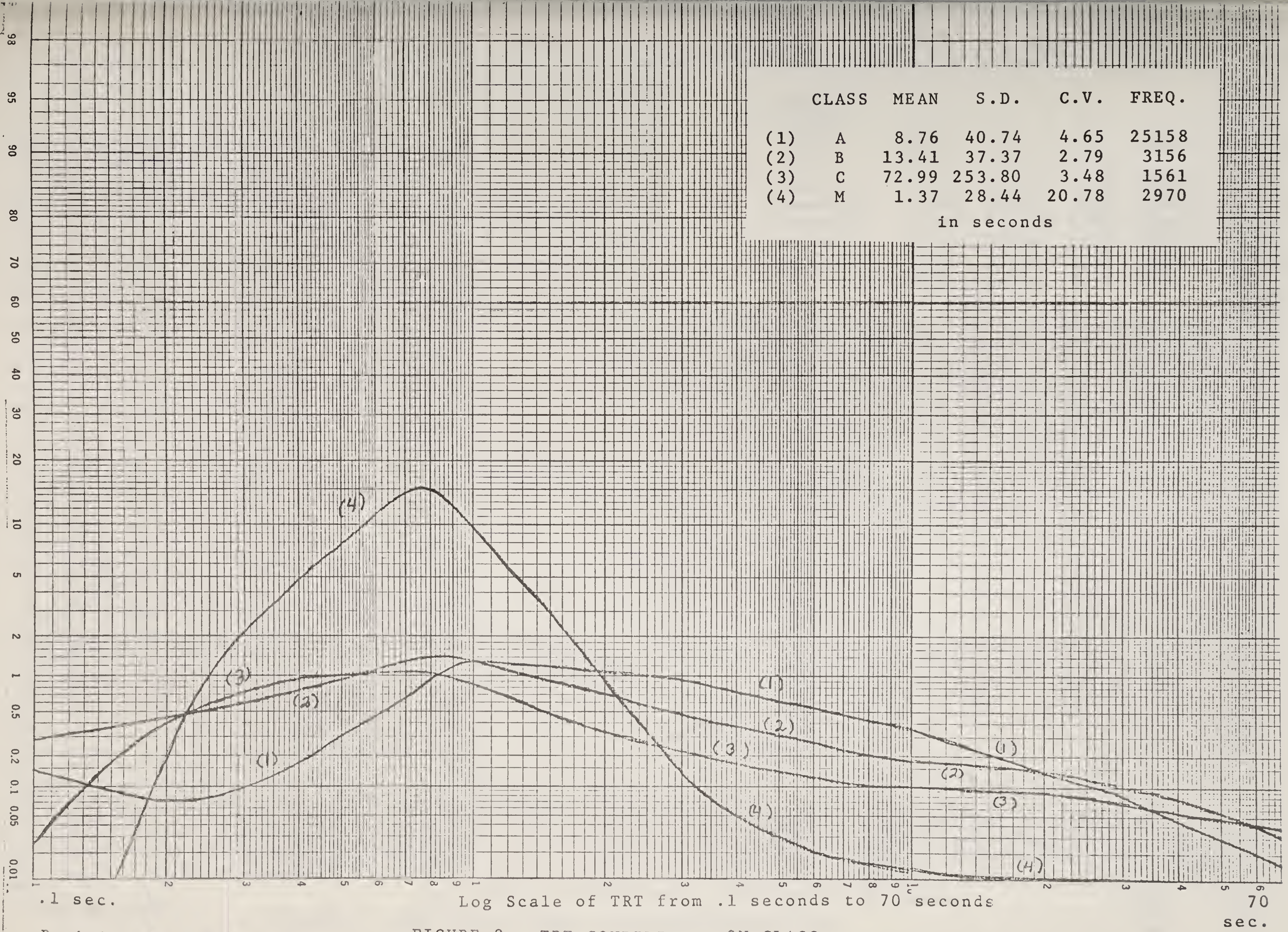


	CLASS	MEAN	S.D.	C.V.	FREQ.
(1)	A	21.51	53.44	2.44	24902
(2)	B	62.24	148.19	2.38	3032
(3)	C	236.03	442.65	1.88	1395
(4)	M	9.04	75.81	8.39	2932

in seconds

Probability
x
Log Paper

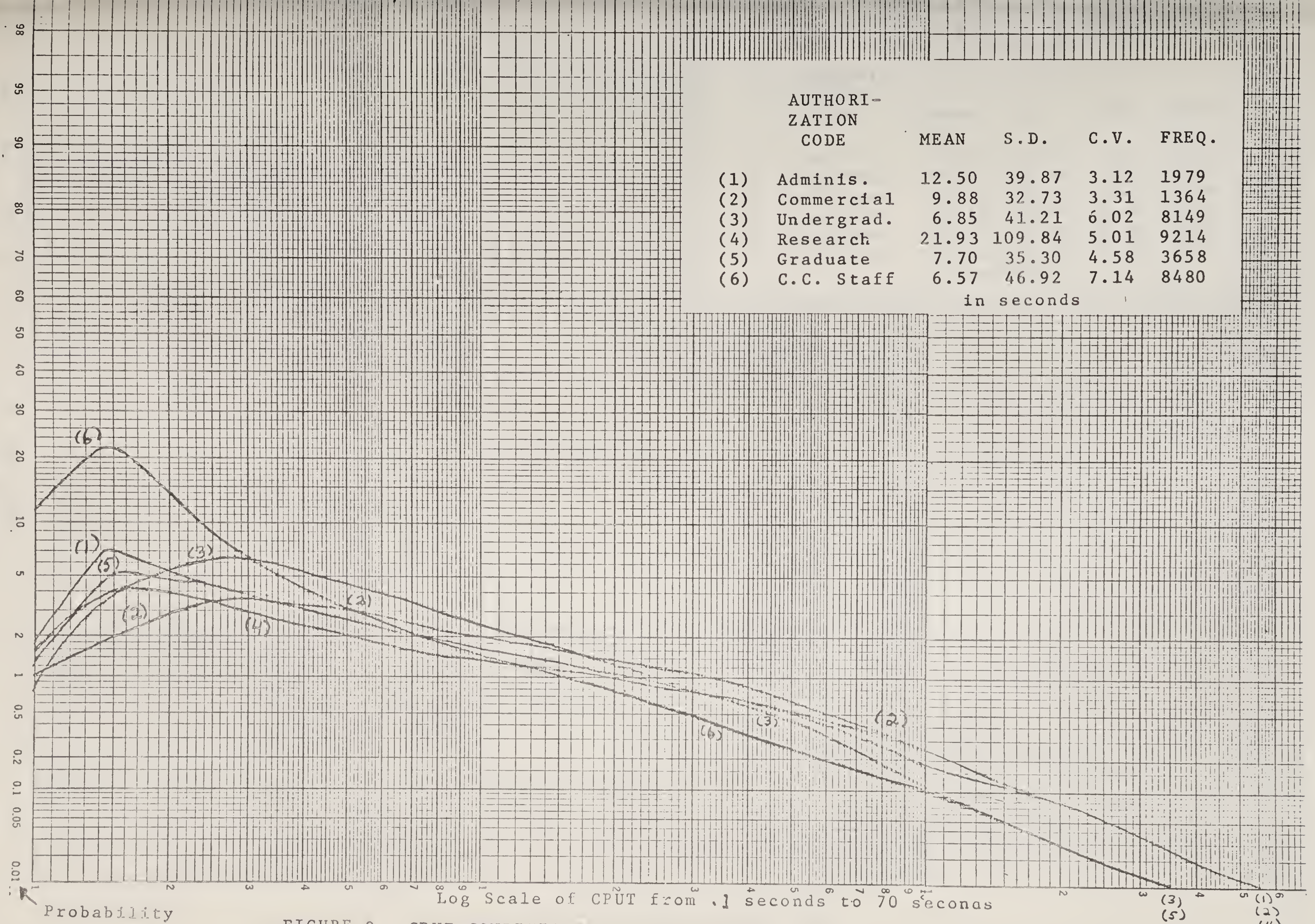
FIGURE 7. CPUT CONDITIONED ON CLASS



	CLASS	MEAN	S.D.	C.V.	FREQ.
(1)	A	8.76	40.74	4.65	25158
(2)	B	13.41	37.37	2.79	3156
(3)	C	72.99	253.80	3.48	1561
(4)	M	1.37	28.44	20.78	2970

in seconds

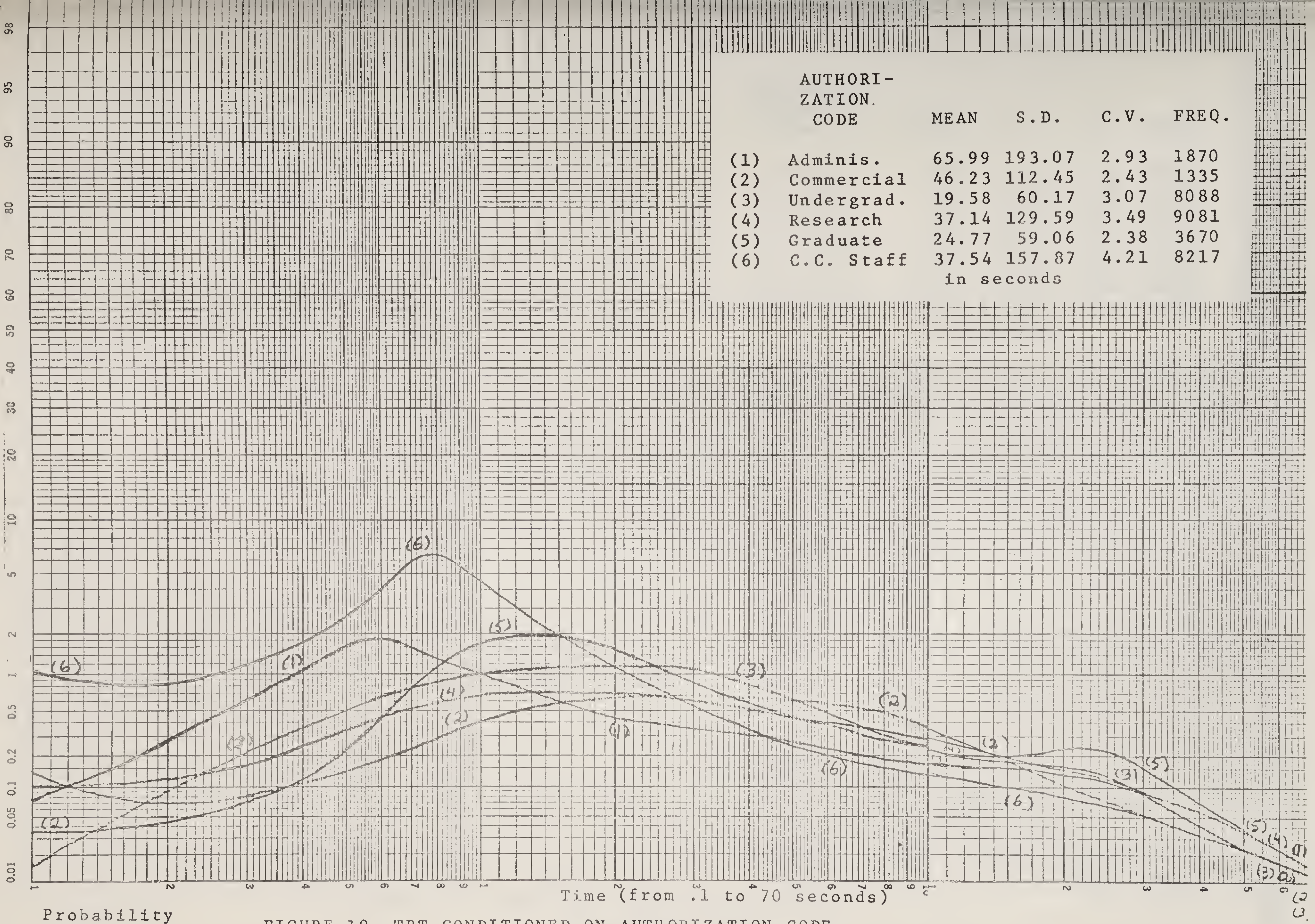
FIGURE 8. TRT CONDITIONED ON CLASS



AUTHORI-ZATION CODE		MEAN	S.D.	C.V.	FREQ.
(1)	Adminis.	12.50	39.87	3.12	1979
(2)	Commercial	9.88	32.73	3.31	1364
(3)	Undergrad.	6.85	41.21	6.02	8149
(4)	Research	21.93	109.84	5.01	9214
(5)	Graduate	7.70	35.30	4.58	3658
(6)	C.C. Staff	6.57	46.92	7.14	8480

in seconds

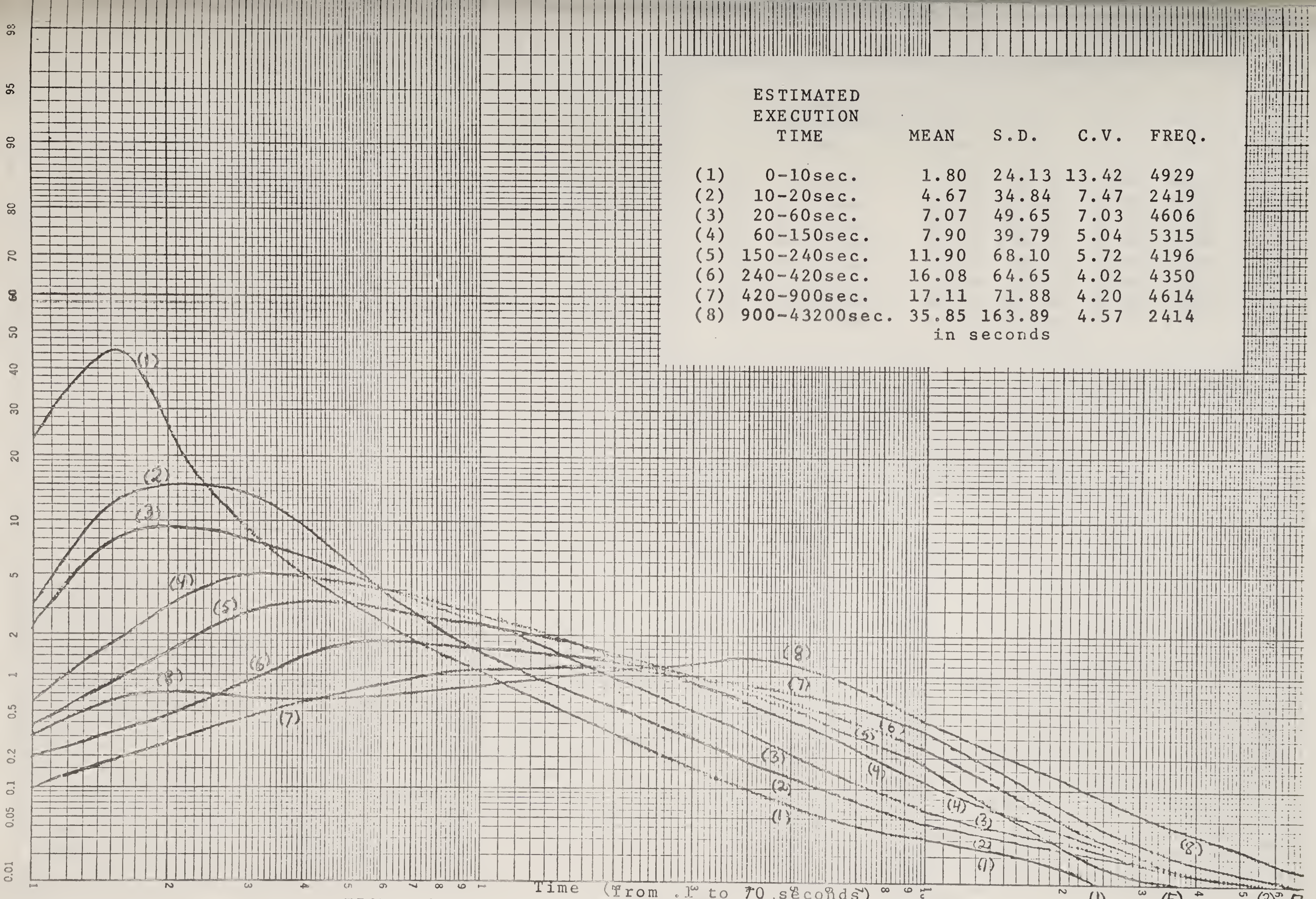
FIGURE 9. CPUT CONDITIONED ON AUTHORIZATION CODE



	AUTHORIZATION CODE	MEAN	S.D.	C.V.	FREQ.
(1)	Adminis.	65.99	193.07	2.93	1870
(2)	Commercial	46.23	112.45	2.43	1335
(3)	Undergrad.	19.58	60.17	3.07	8088
(4)	Research	37.14	129.59	3.49	9081
(5)	Graduate	24.77	59.06	2.38	3670
(6)	C.C. Staff	37.54	157.87	4.21	8217

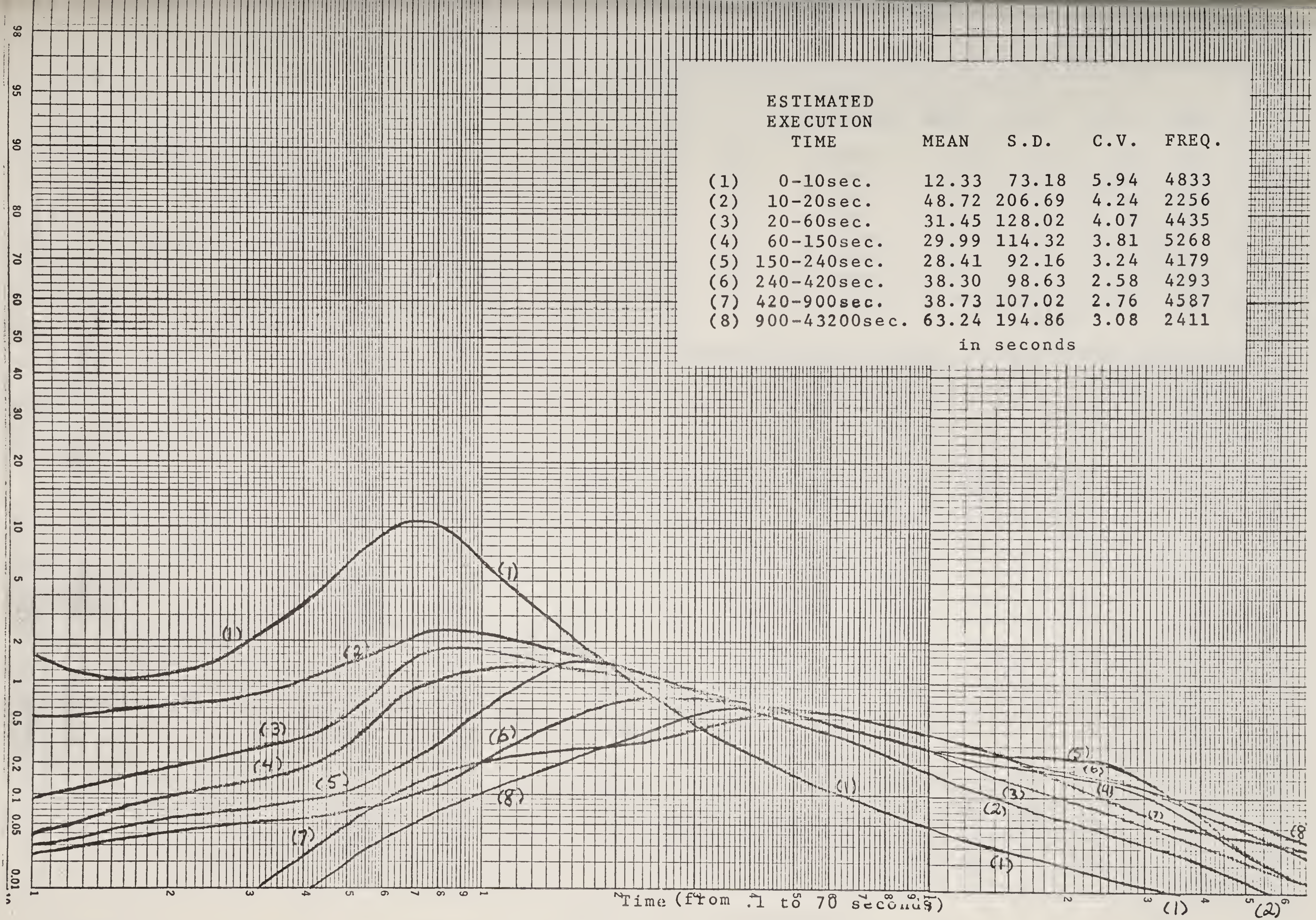
in seconds

FIGURE 10. TRT CONDITIONED ON AUTHORIZATION CODE



ESTIMATED EXECUTION TIME		MEAN	S.D.	C.V.	FREQ.
(1)	0-10sec.	1.80	24.13	13.42	4929
(2)	10-20sec.	4.67	34.84	7.47	2419
(3)	20-60sec.	7.07	49.65	7.03	4606
(4)	60-150sec.	7.90	39.79	5.04	5315
(5)	150-240sec.	11.90	68.10	5.72	4196
(6)	240-420sec.	16.08	64.65	4.02	4350
(7)	420-900sec.	17.11	71.88	4.20	4614
(8)	900-43200sec.	35.85	163.89	4.57	2414
		in seconds			

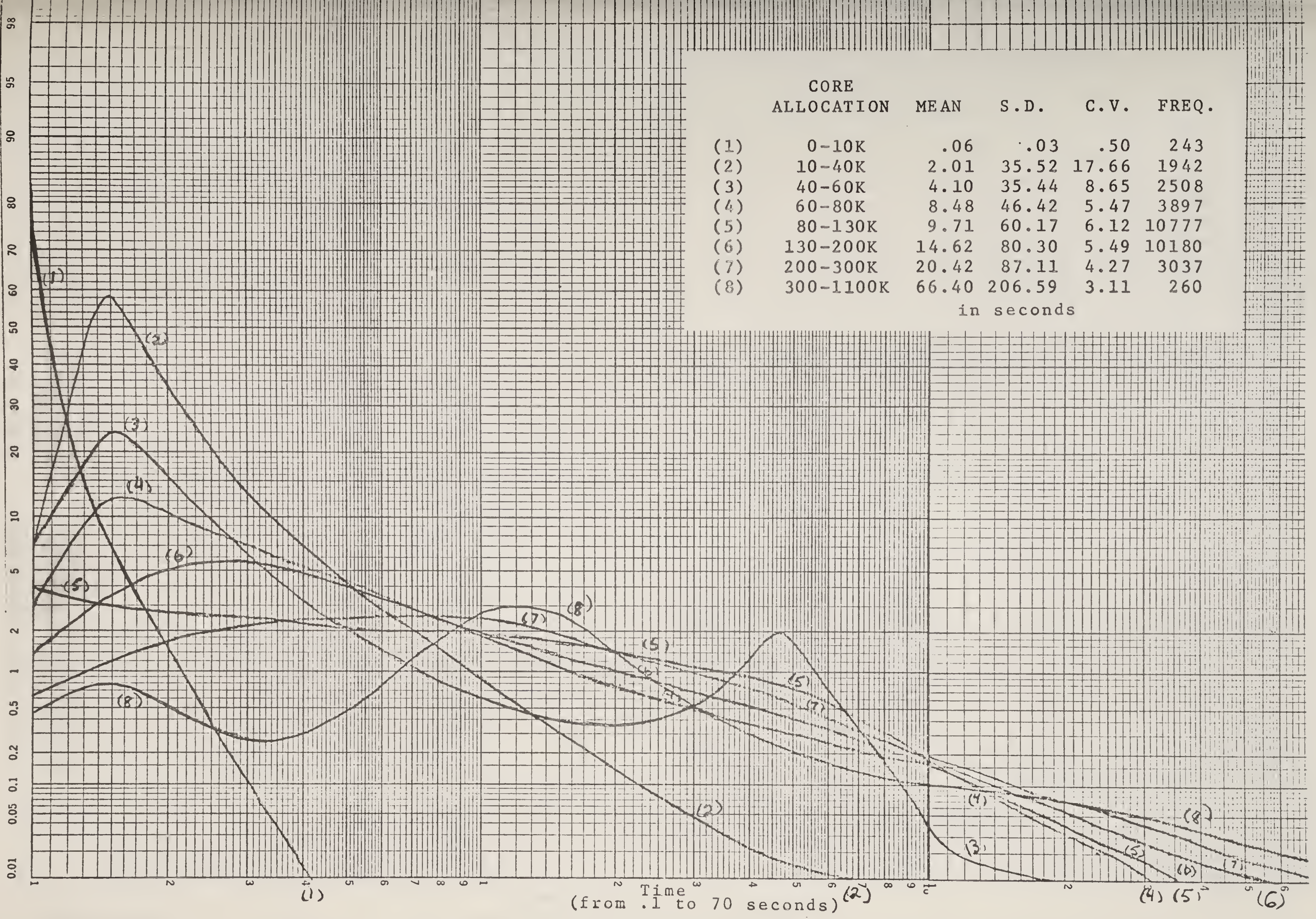
FIGURE 11. CPUT CONDITIONED ON ESTIMATED EXECUTION TIME



	ESTIMATED EXECUTION TIME	MEAN	S.D.	C.V.	FREQ.
(1)	0-10sec.	12.33	73.18	5.94	4833
(2)	10-20sec.	48.72	206.69	4.24	2256
(3)	20-60sec.	31.45	128.02	4.07	4435
(4)	60-150sec.	29.99	114.32	3.81	5268
(5)	150-240sec.	28.41	92.16	3.24	4179
(6)	240-420sec.	38.30	98.63	2.58	4293
(7)	420-900sec.	38.73	107.02	2.76	4587
(8)	900-43200sec.	63.24	194.86	3.08	2411

in seconds

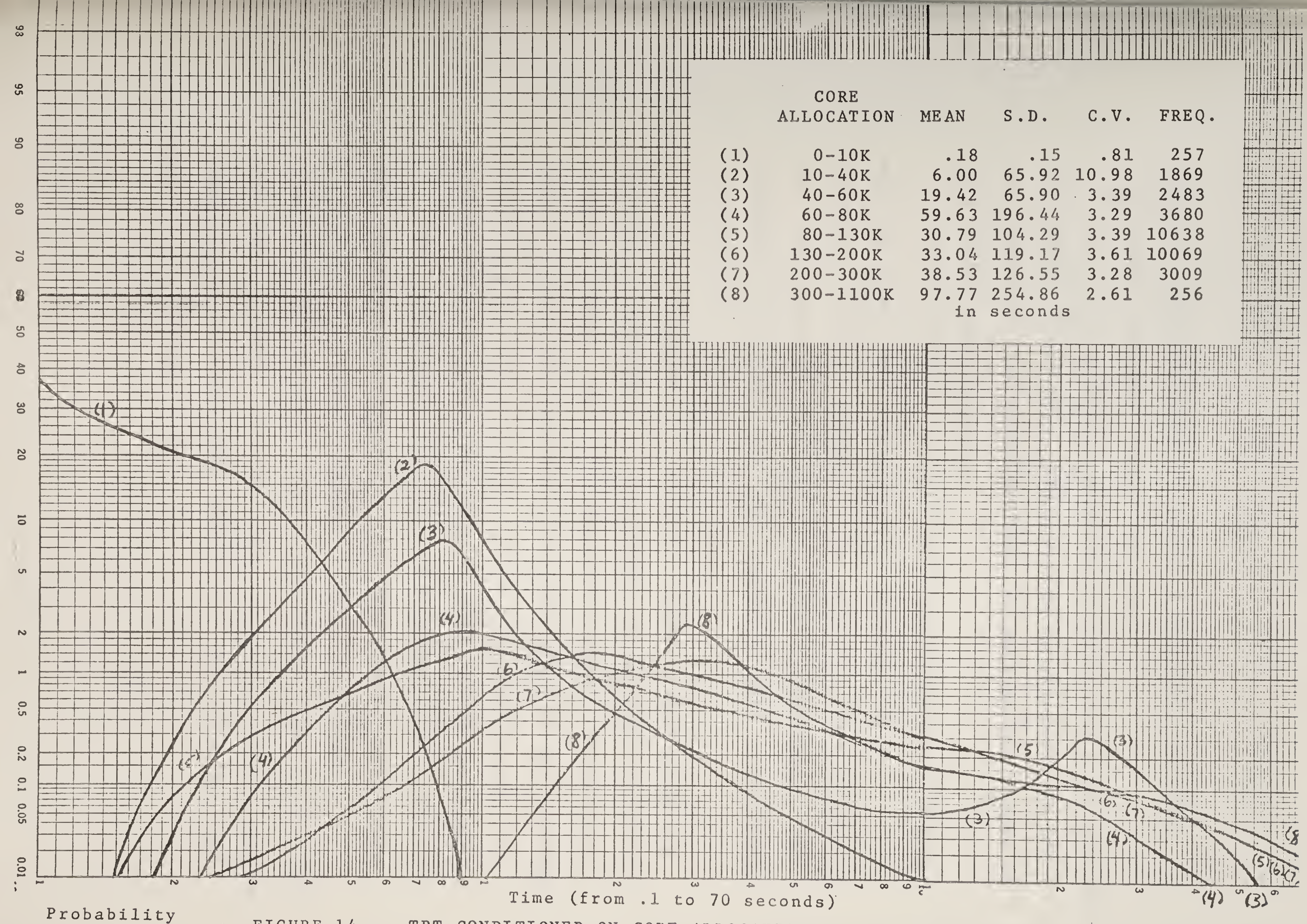
FIGURE 12. TRT CONDITIONED ON ESTIMATED EXECUTION TIME



	CORE ALLOCATION	MEAN	S.D.	C.V.	FREQ.
(1)	0-10K	.06	.03	.50	243
(2)	10-40K	2.01	35.52	17.66	1942
(3)	40-60K	4.10	35.44	8.65	2508
(4)	60-80K	8.48	46.42	5.47	3897
(5)	80-130K	9.71	60.17	6.12	10777
(6)	130-200K	14.62	80.30	5.49	10180
(7)	200-300K	20.42	87.11	4.27	3037
(8)	300-1100K	66.40	206.59	3.11	260

in seconds

FIGURE 13. CPUT CONDITIONED ON CORE ALLOCATION



	CORE ALLOCATION	MEAN	S.D.	C.V.	FREQ.
(1)	0-10K	.18	.15	.81	257
(2)	10-40K	6.00	65.92	10.98	1869
(3)	40-60K	19.42	65.90	3.39	2483
(4)	60-80K	59.63	196.44	3.29	3680
(5)	80-130K	30.79	104.29	3.39	10638
(6)	130-200K	33.04	119.17	3.61	10069
(7)	200-300K	38.53	126.55	3.28	3009
(8)	300-1100K	97.77	254.86	2.61	256

in seconds

FIGURE 14. TRT CONDITIONED ON CORE ALLOCATION

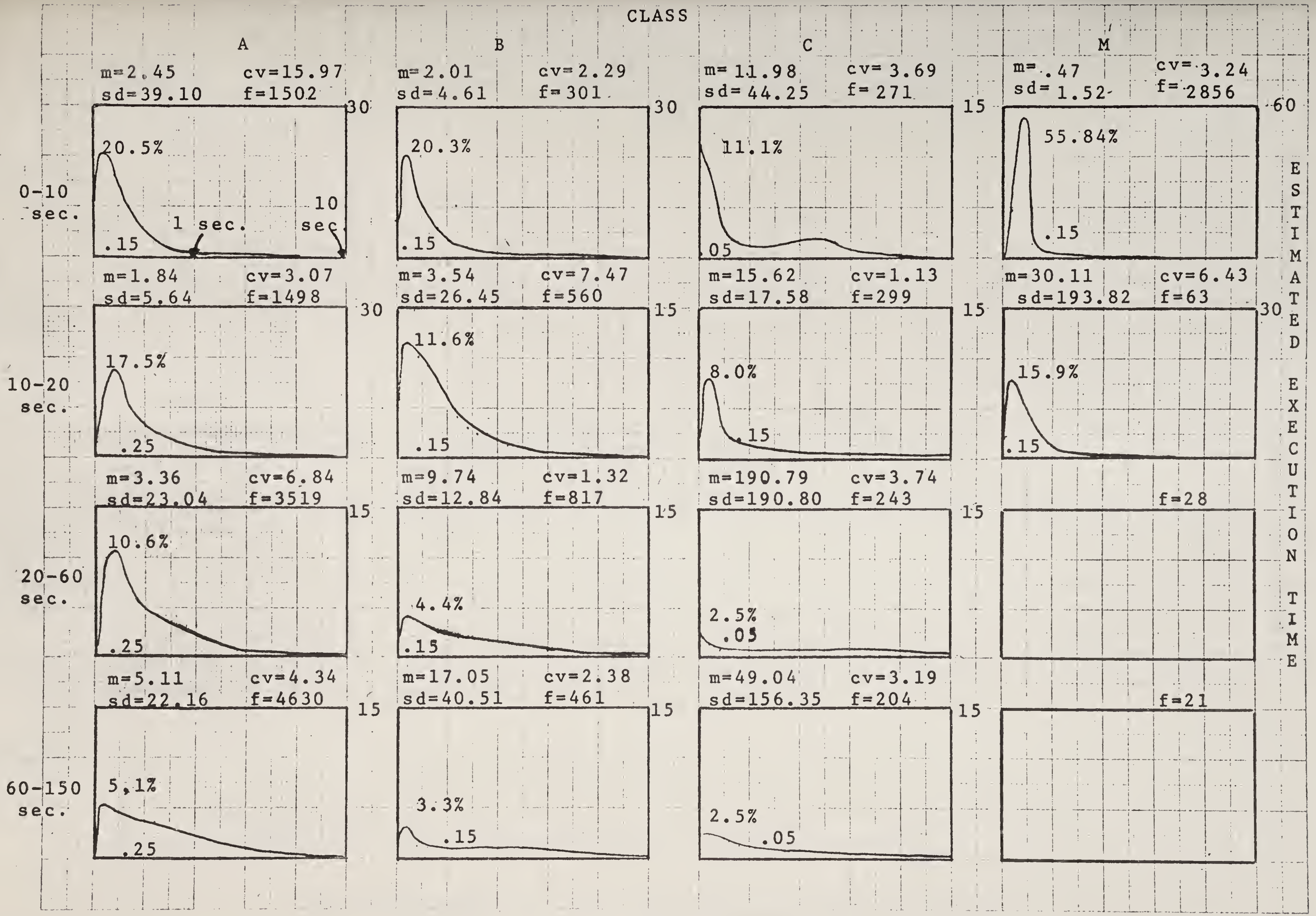
shape of the distributions, not the details. In this regard, the following should be noted:

(1) The ordinates vary from 0 to 15, to 0 to 100, and are marked accordingly. This method is necessary to convey the shapes of flat distributions, as well as the shapes of very peaked distributions.

(2) The first two-fifths of the abscissa is divided equally among the intervals, 0-.1 seconds, .1-.2 seconds,...,.9-1.0 seconds. The remaining three-fifths of the abscissa is divided equally among the intervals, 1-4 seconds, 4-7 seconds, and 7-10 seconds. This is necessary to convey the nature of the peaks.

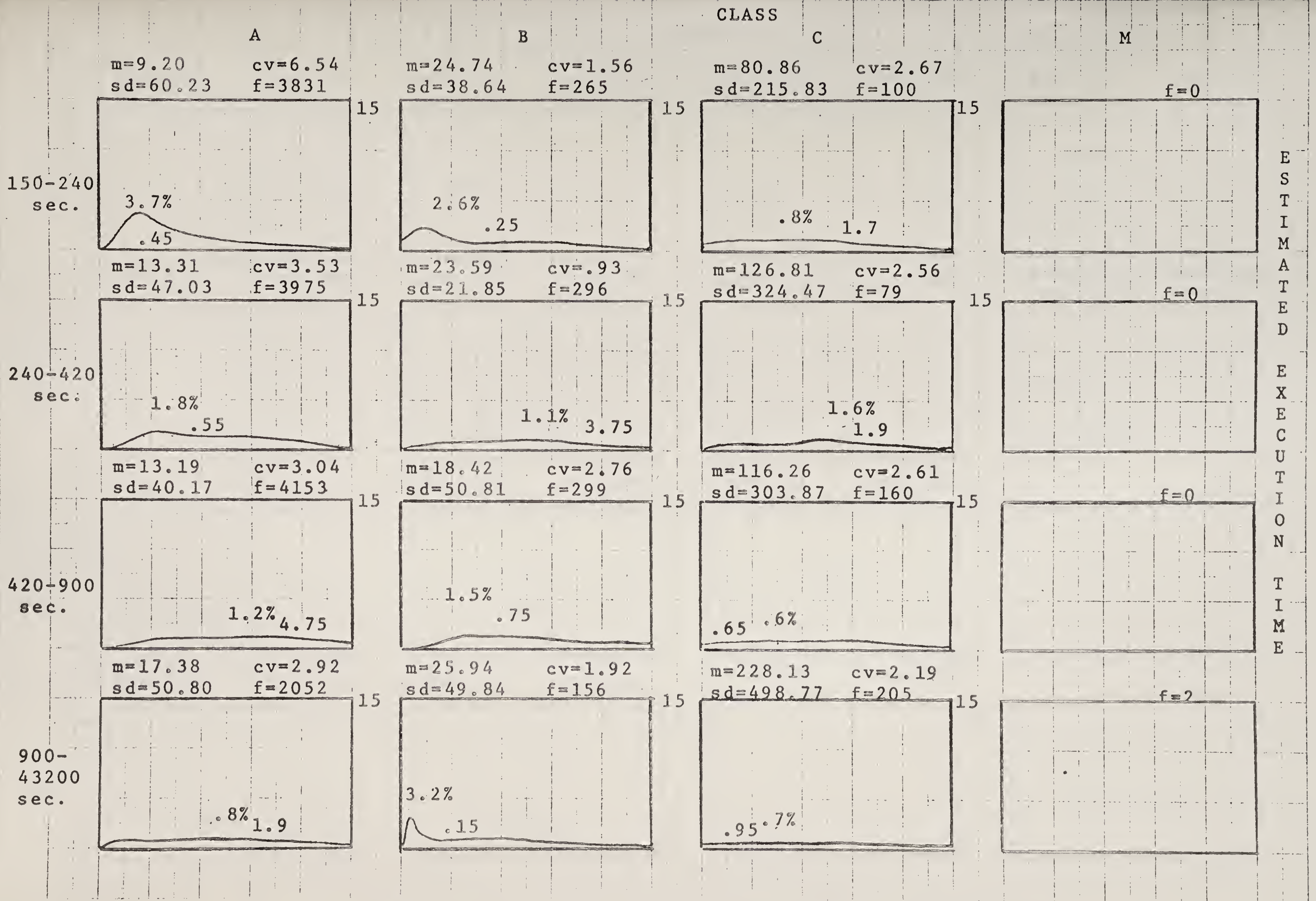
(3) The mean (in seconds), standard deviation (in seconds), coefficient of variation, and frequency are given above the graph, for that particular graph. On the graph itself, appear the percentage of jobs lying in the interval containing the maximum point of the distribution, and the maximum point value in relation to CPUT or TRT (in seconds).

(4) Occasionally, a box will appear blank, with only the frequency provided above it. This class contains too few and too scattered a number of jobs, to provide a meaningful distribution. If a distribution does appear with only the frequency above it, then the class contains too few jobs to provide meaningful statistics, but those few are concentrated in one area.

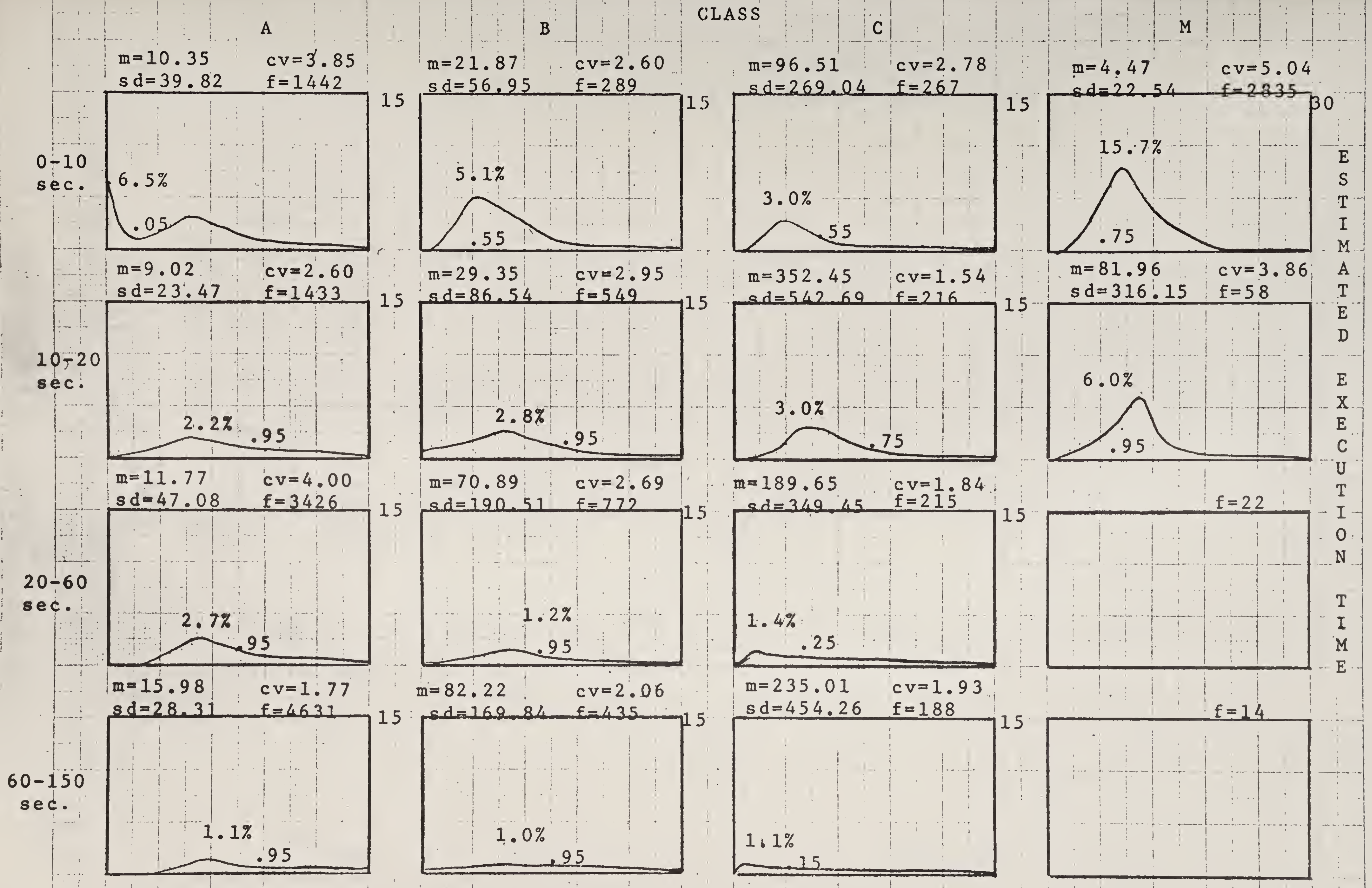


CPUT CONDITIONED ON CLASS AND ESTIMATED EXECUTION TIME
FIGURE 15.

Page 1 of 2

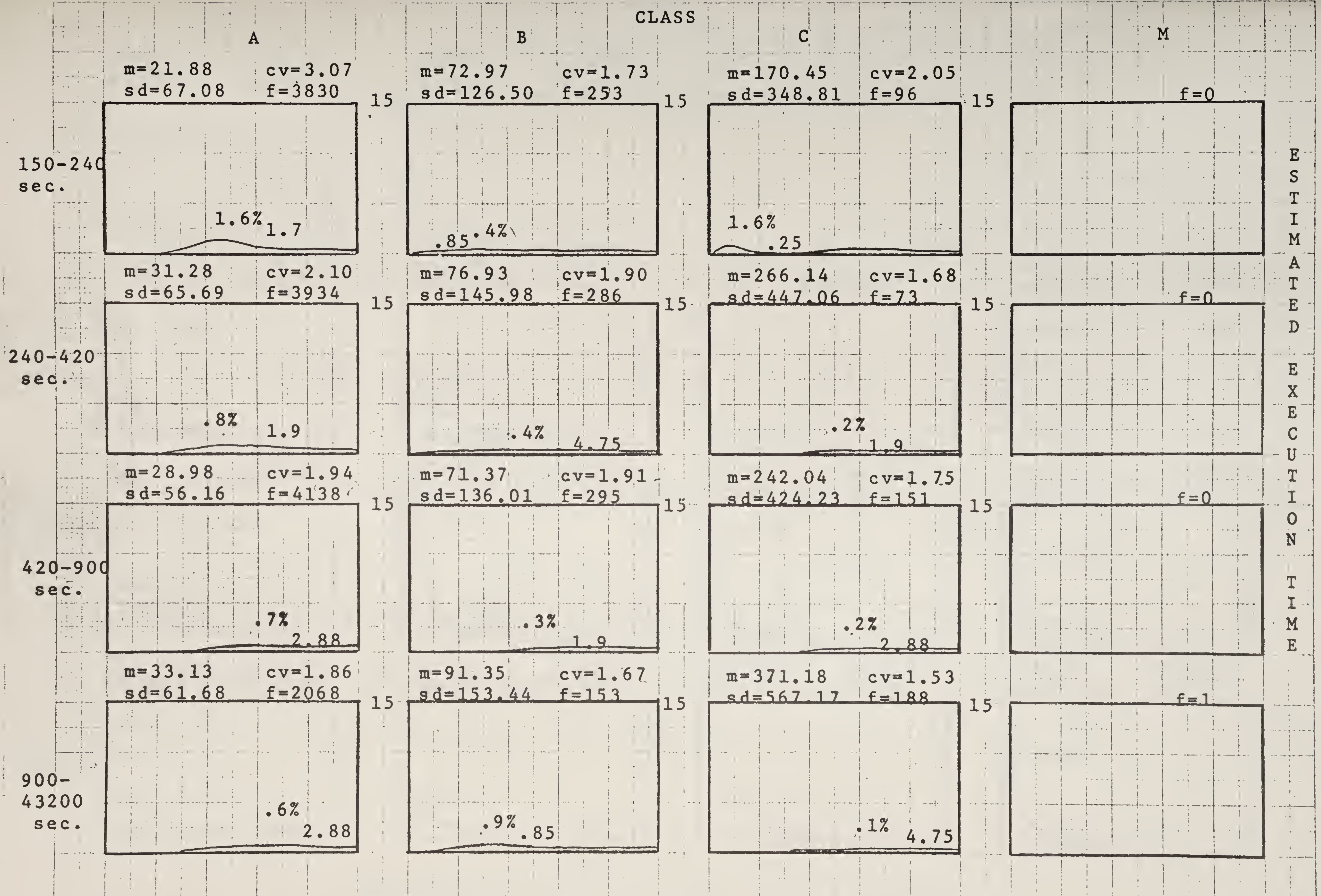


CPUT CONDITIONED ON CLASS AND ESTIMATED EXECUTION TIME



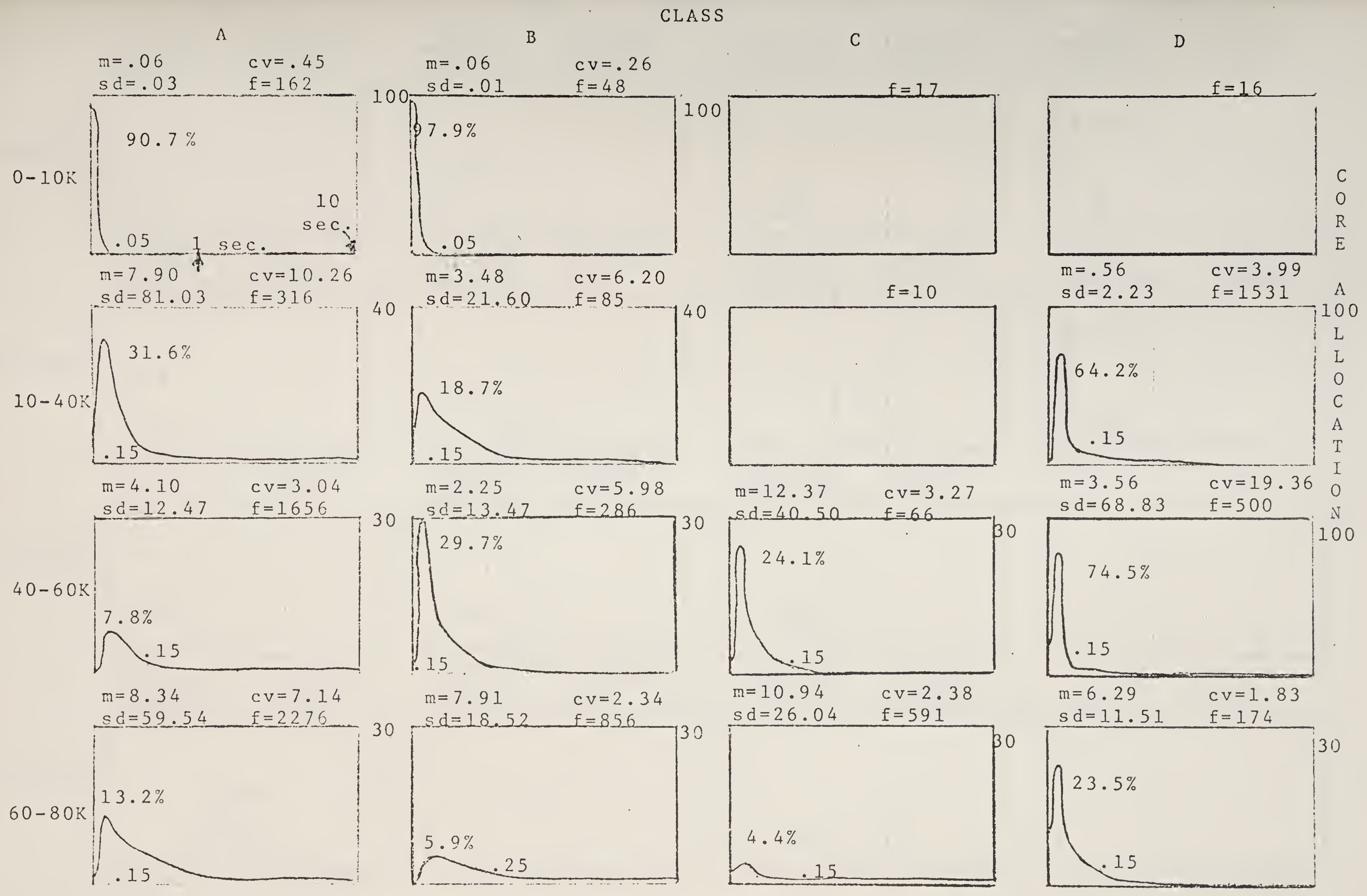
TRT CONDITIONED ON CLASS AND ESTIMATED EXECUTION TIME
FIGURE 16.

Page 1 of 2



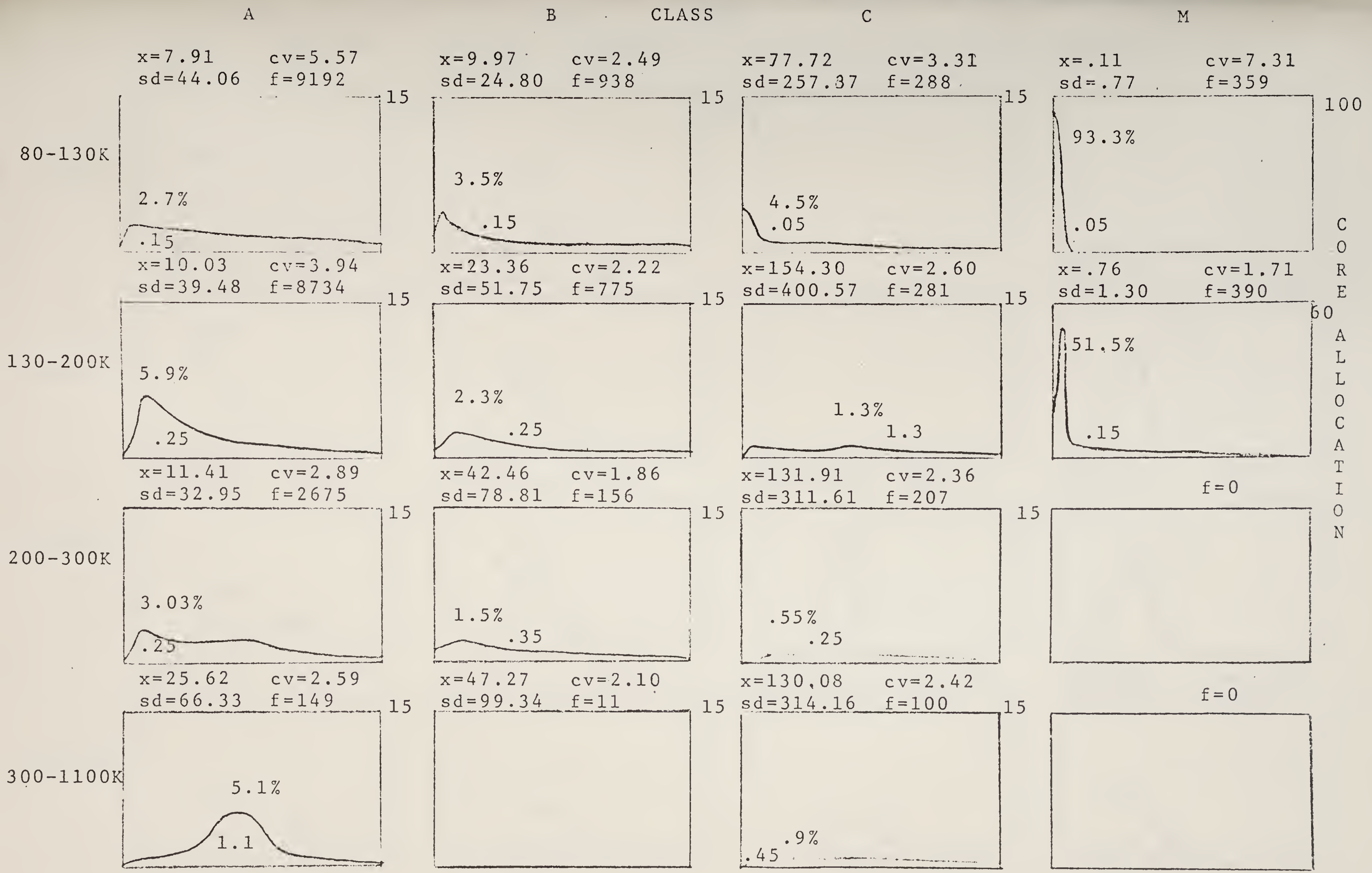
TRT CONDITIONED ON CLASS AND ESTIMATED EXECUTION TIME

Page 2 of 2



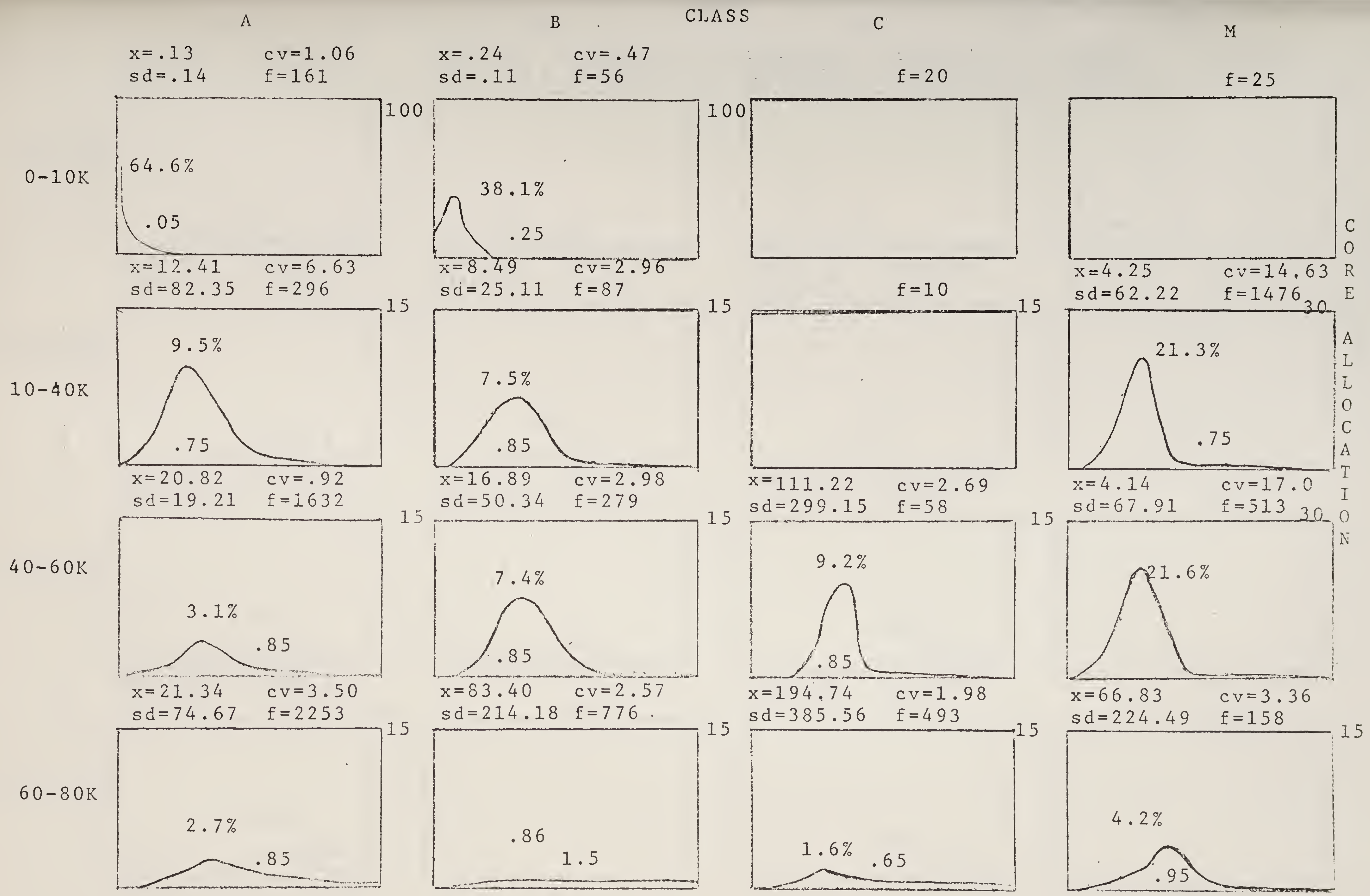
CPUT CONDITIONED ON CLASS AND CORE ALLOCATION
FIGURE 17.

Page 1 of 2



CPUT CONDITIONED ON CLASS AND CORE ALLOCATION

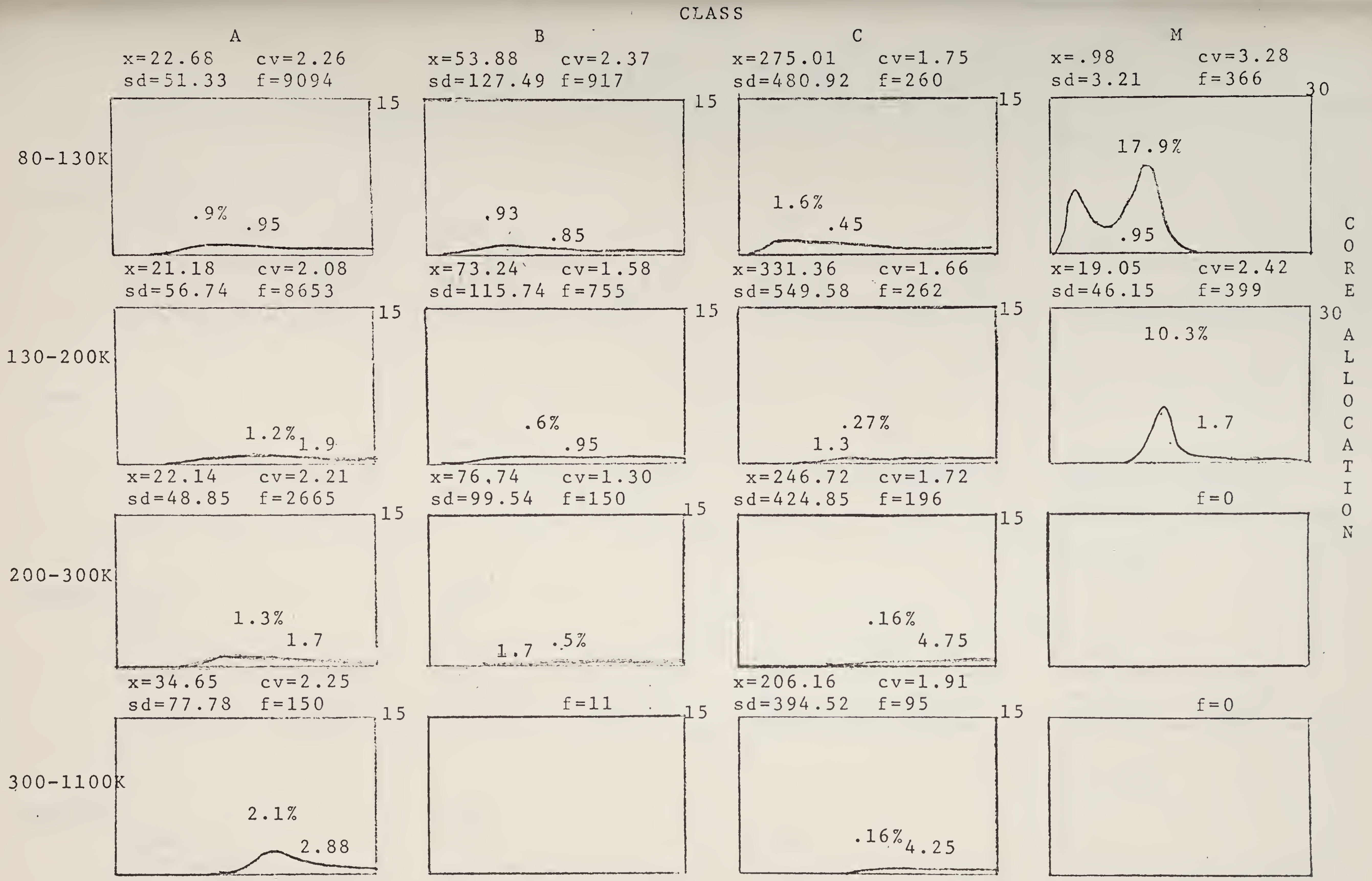
Page 2 of 2



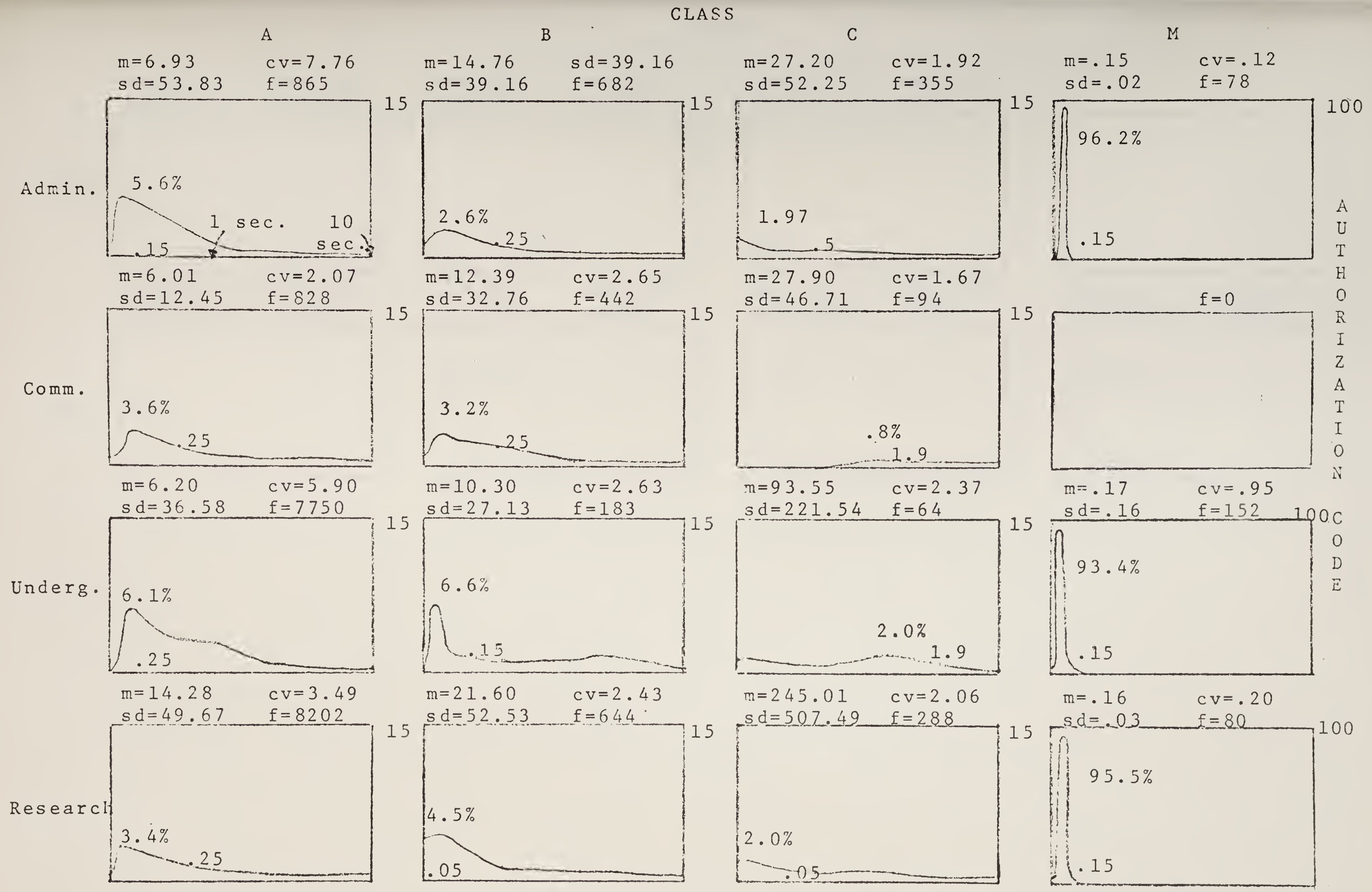
TRT CONDITIONED ON CLASS AND CORE ALLOCATION

FIGURE 18.

Page 1 of 2

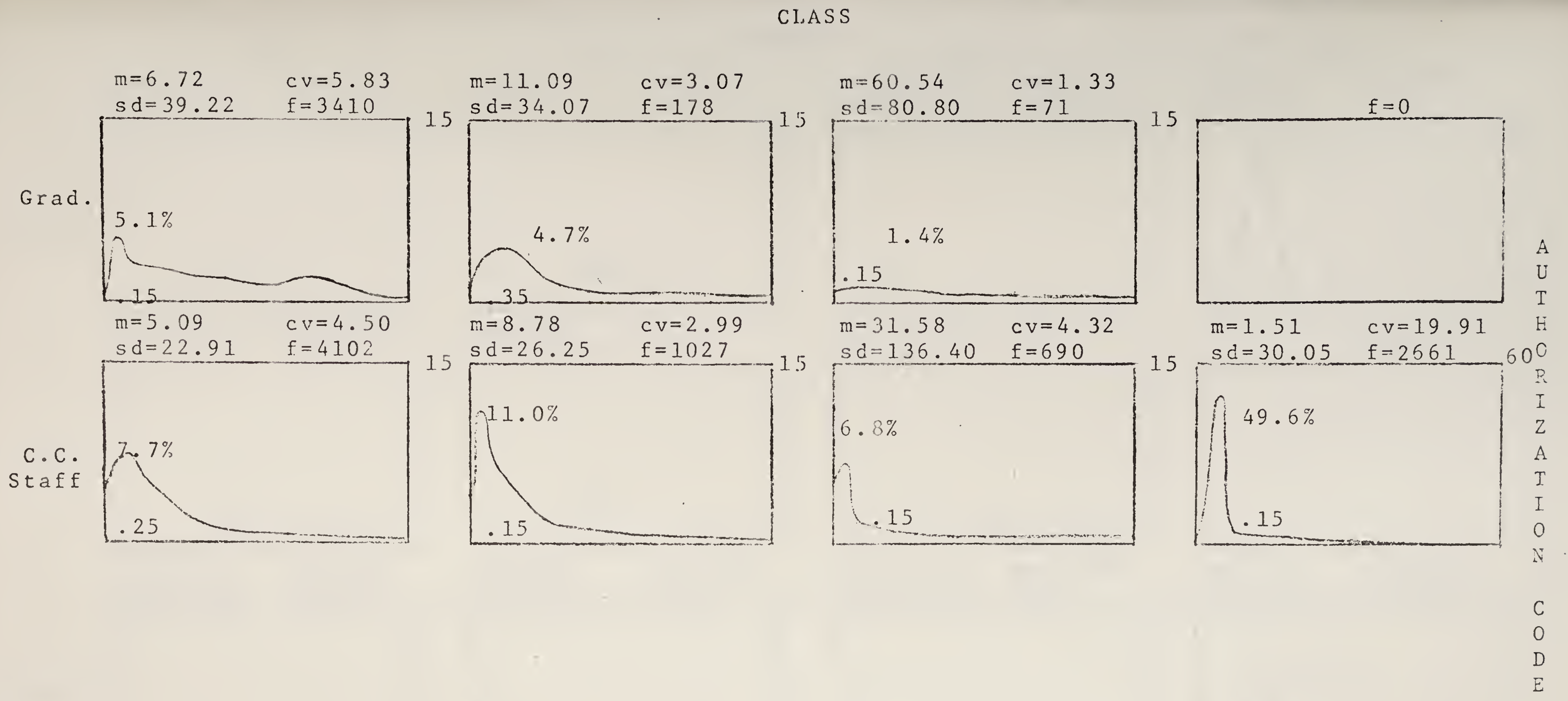


TRT CONDITIONED ON CLASS AND CORE ALLOCATION

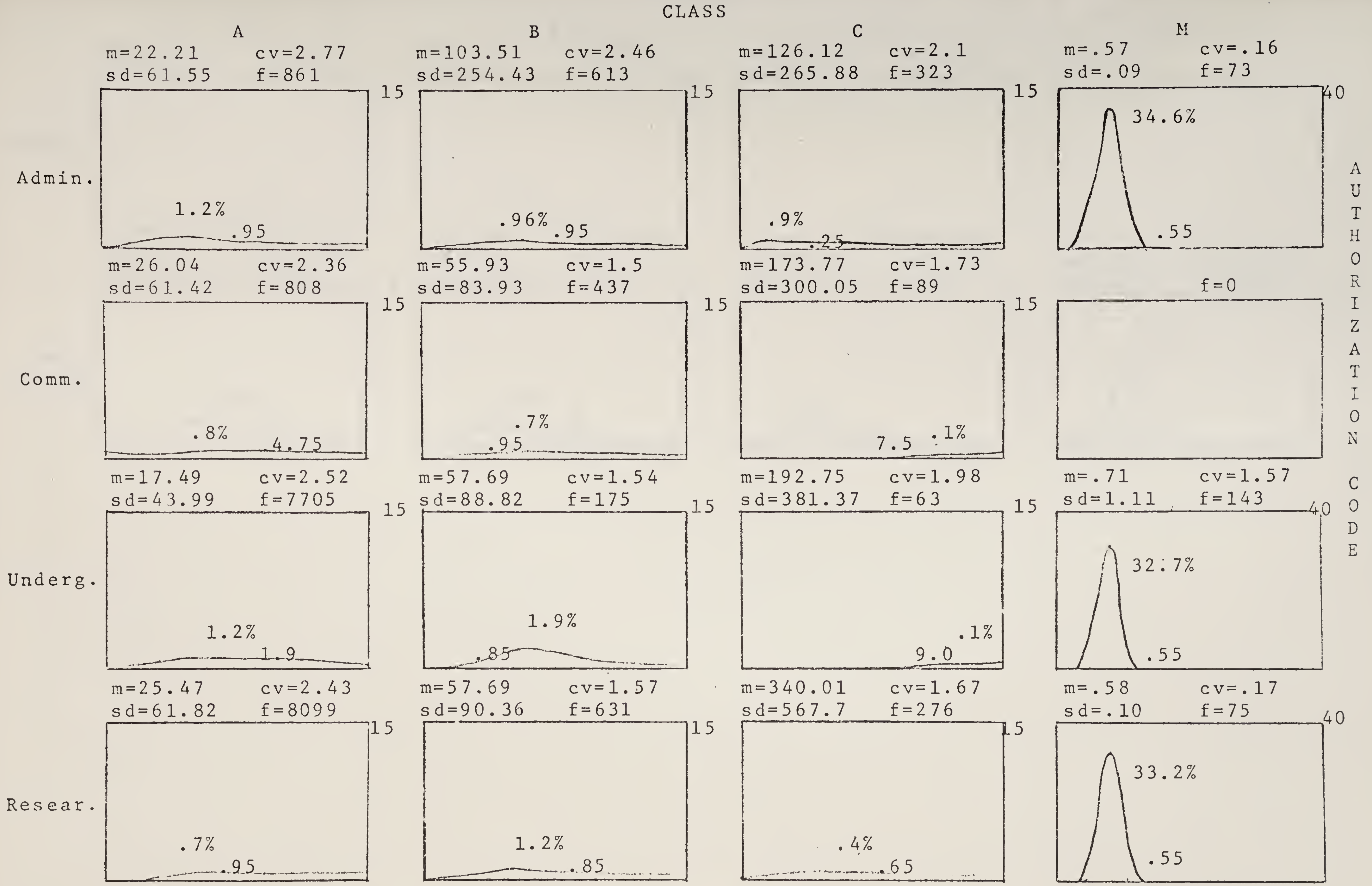


CPUT CONDITIONED ON CLASS AND AUTHORIZATION CODE
FIGURE 19.

Page 1 of 2

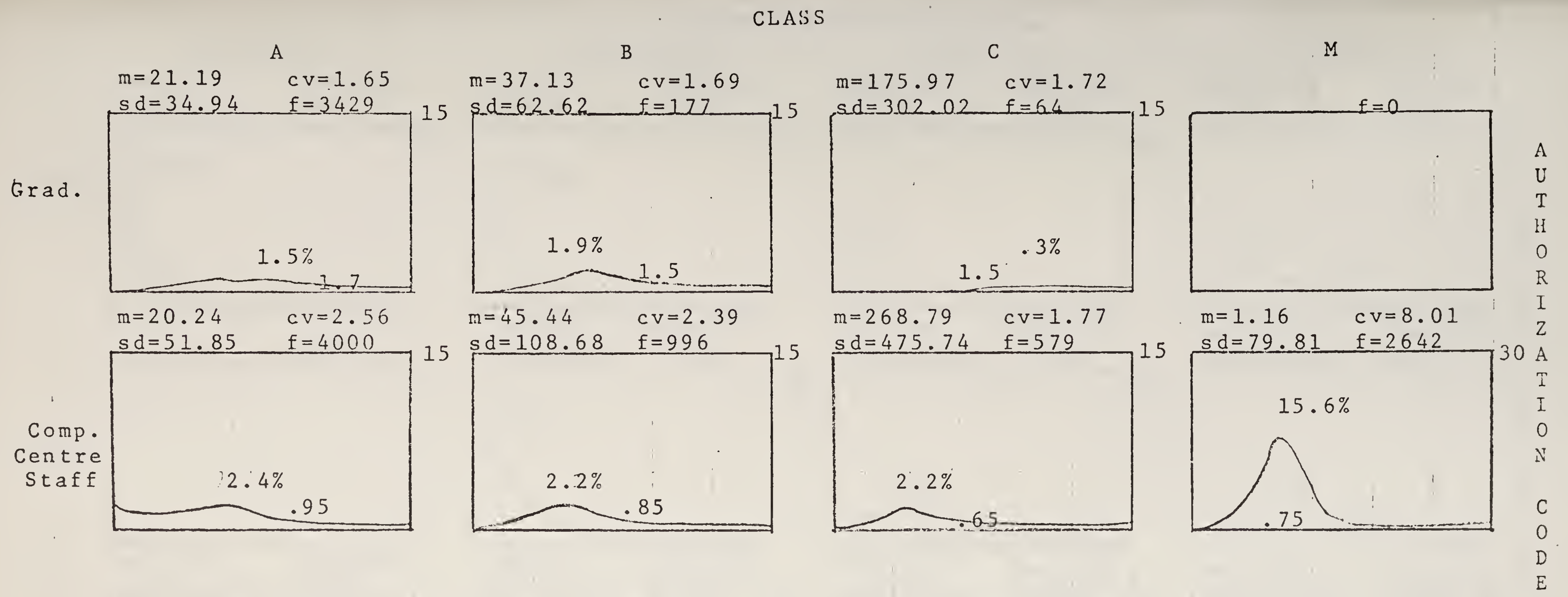


CPUT CONDITIONED ON CLASS AND AUTHORIZATION CODE

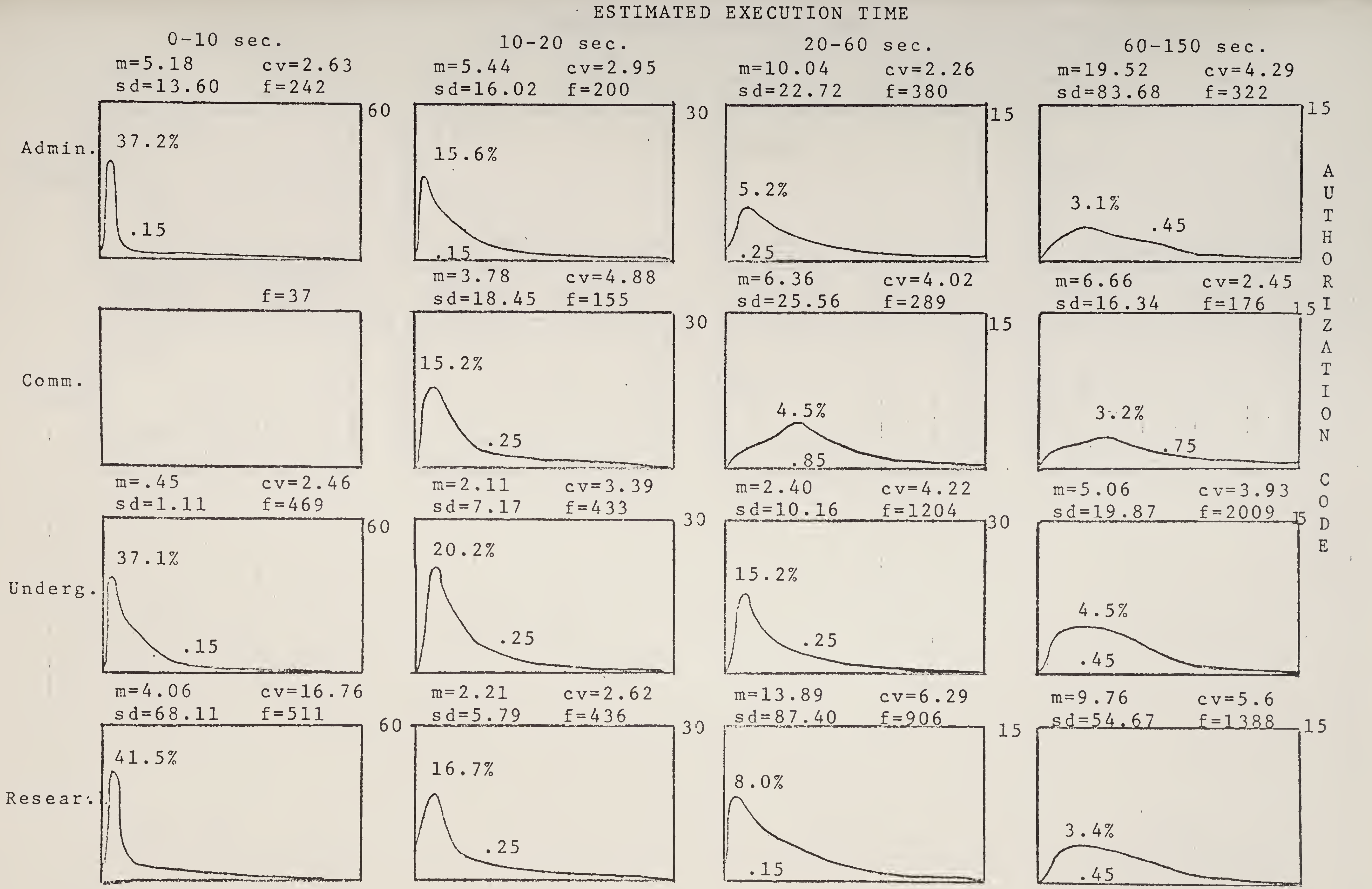


TRT CONDITIONED ON CLASS AND AUTHORIZATION CODE
FIGURE 20.

Page 1 of 2

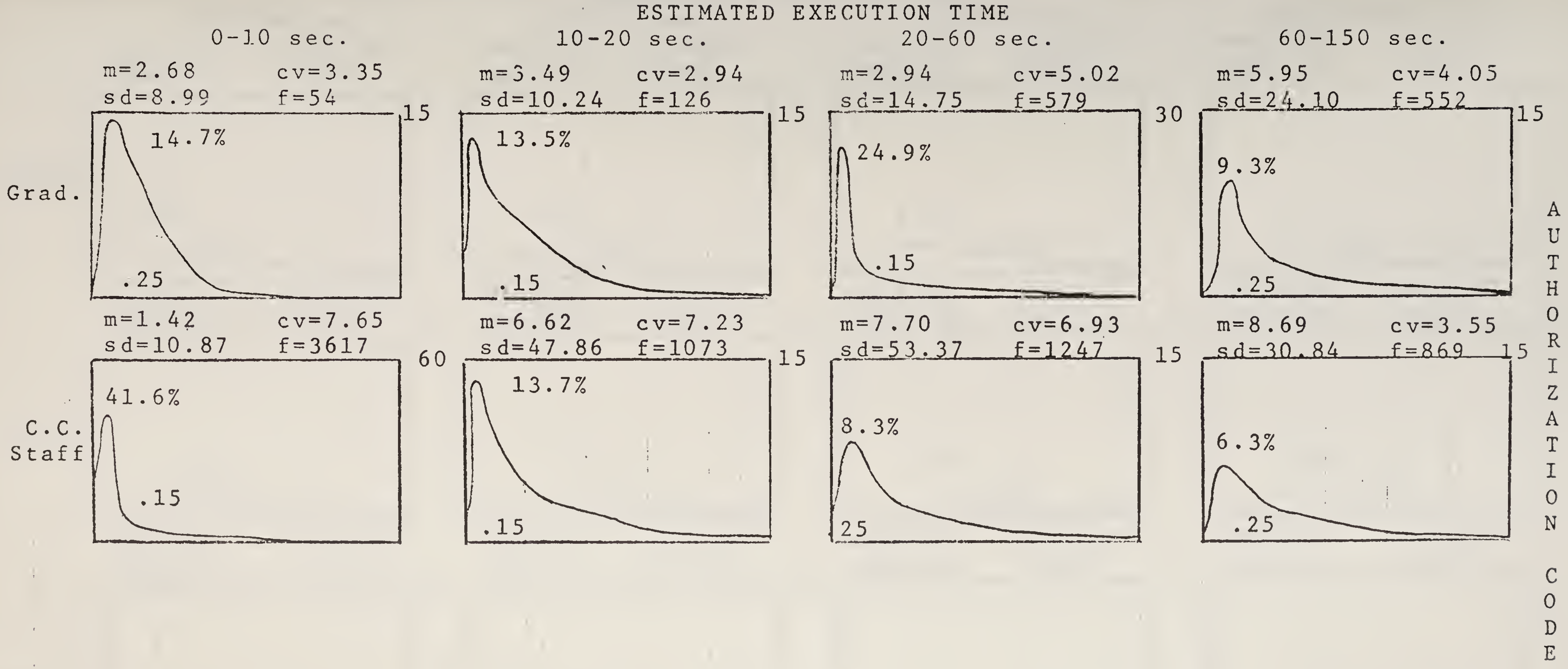


TRT CONDITIONED ON CLASS AND AUTHORIZATION CODE



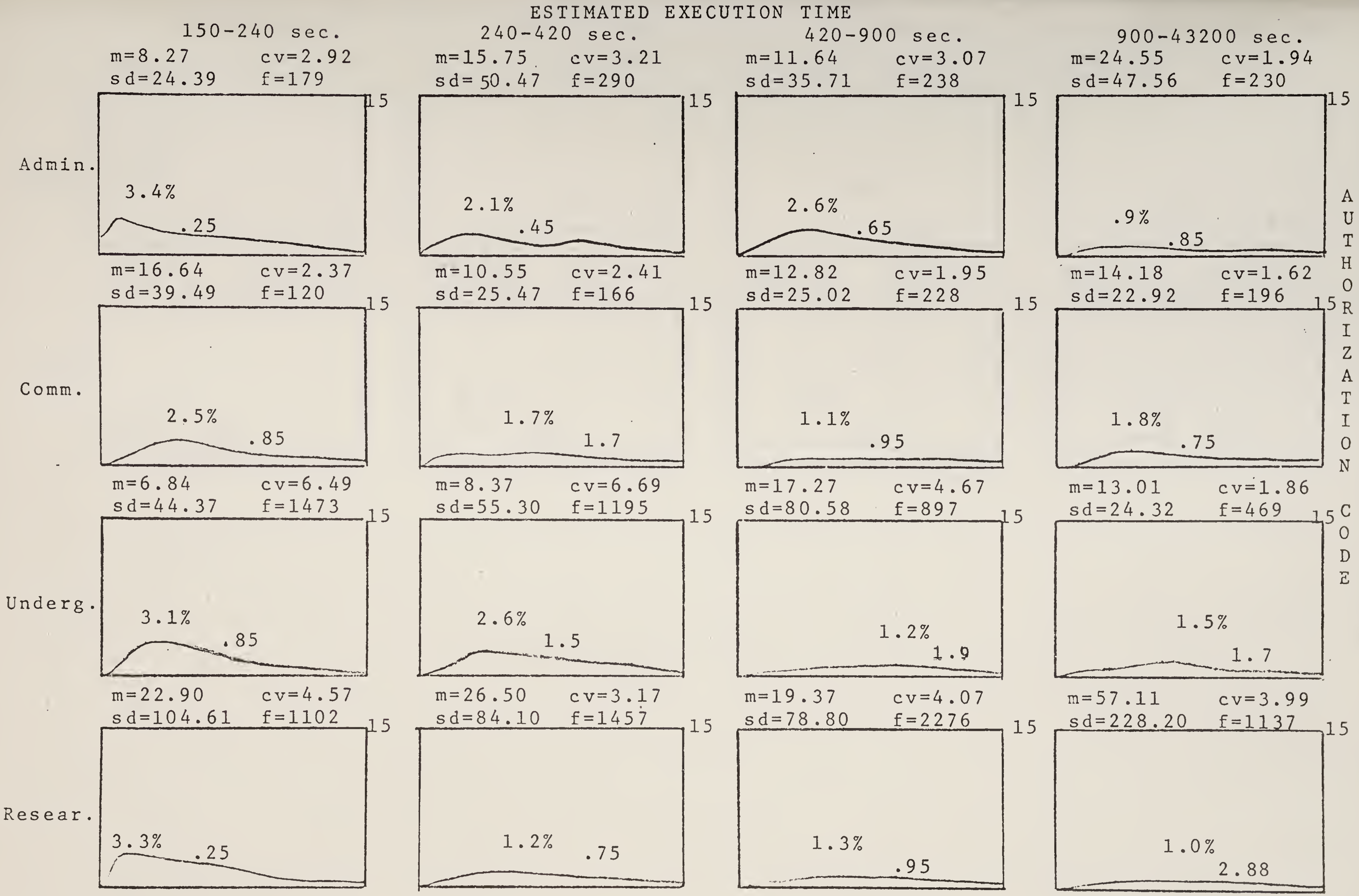
CPUT CONDITIONED ON ESTIMATED EXECUTION TIME AND AUTHORIZATION CODE
FIGURE 21.

Page 1 of 4

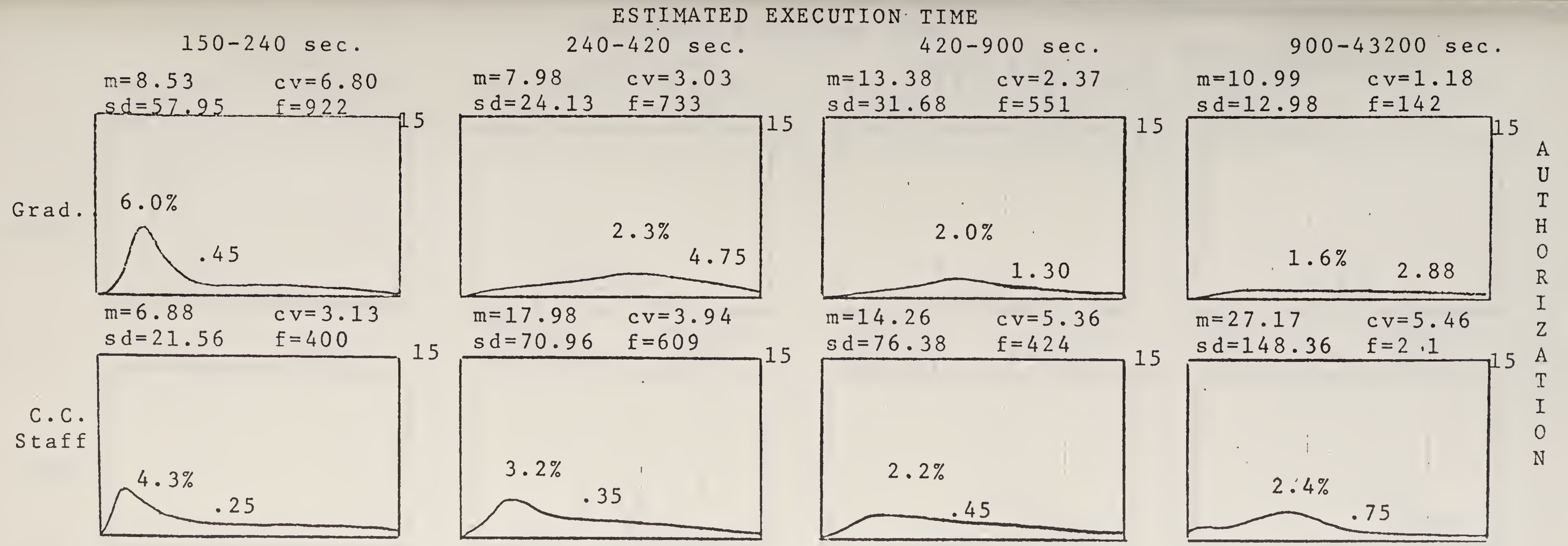


CPUT CONDITIONED ON ESTIMATED EXECUTION TIME AND AUTHORIZATION CODE

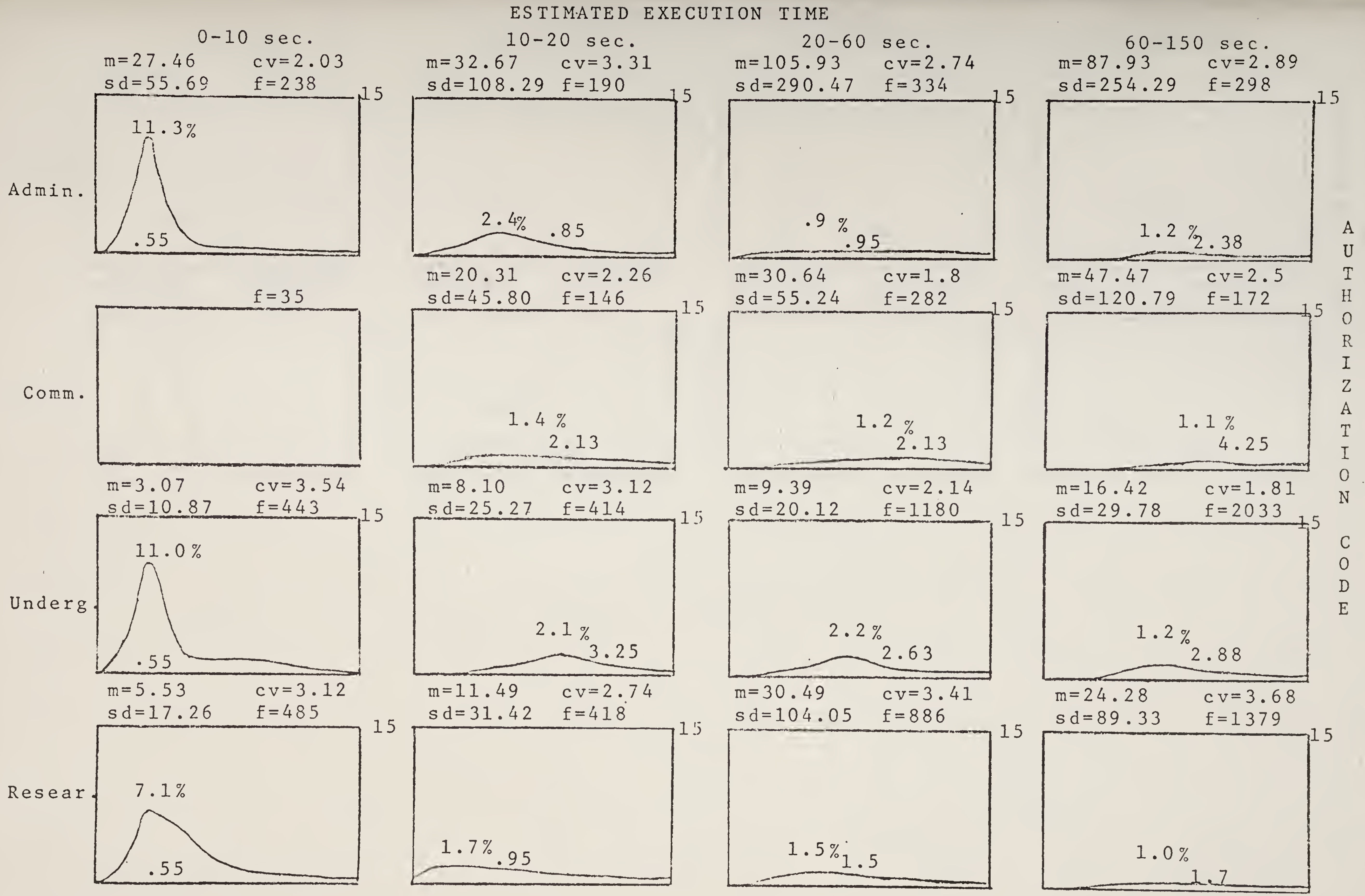
Page 2 of 4



CPUT CONDITIONED ON ESTIMATED EXECUTION TIME AND AUTHORIZATION CODE



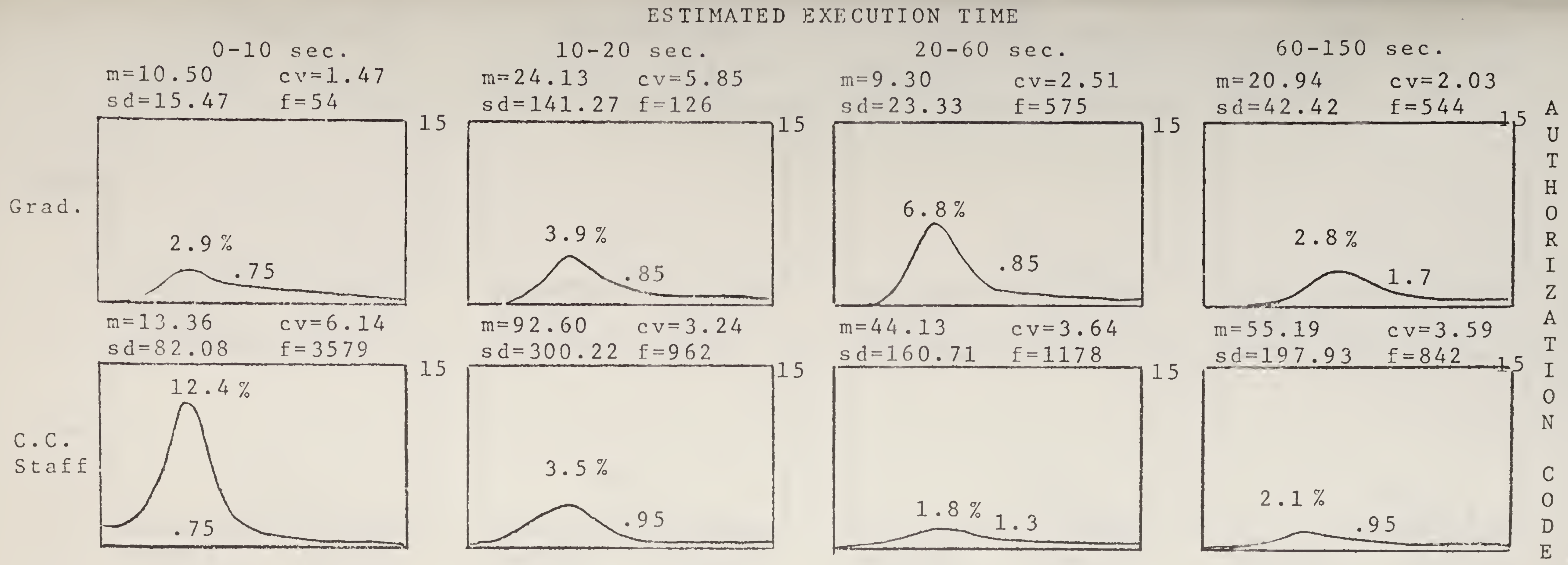
CPUT CONDITIONED ON ESTIMATED EXECUTION TIME AND AUTHORIZATION CODE



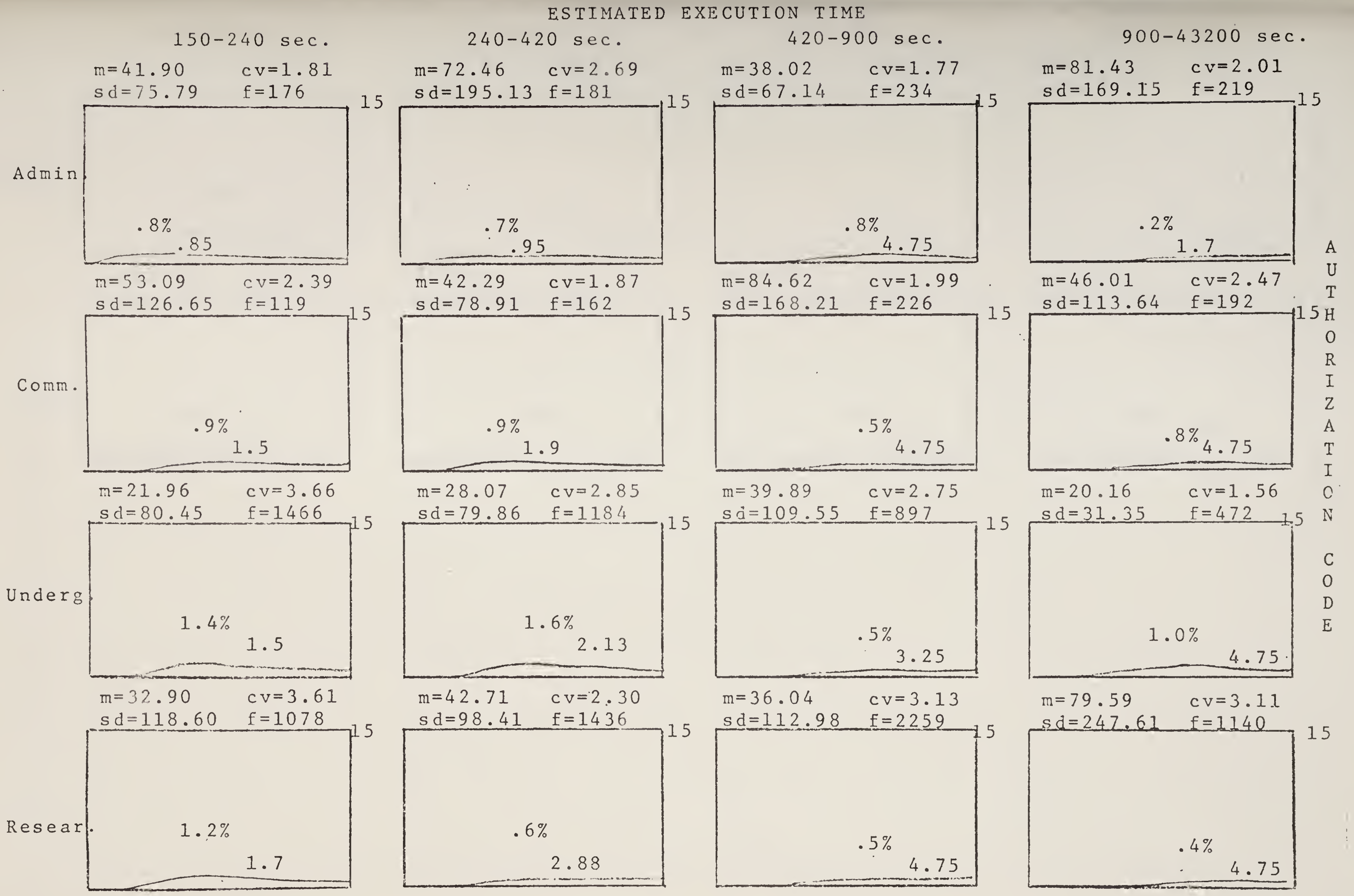
TRT CONDITIONED ON ESTIMATED EXECUTION TIME AND AUTHORIZATION CODE

Page 1 of 4

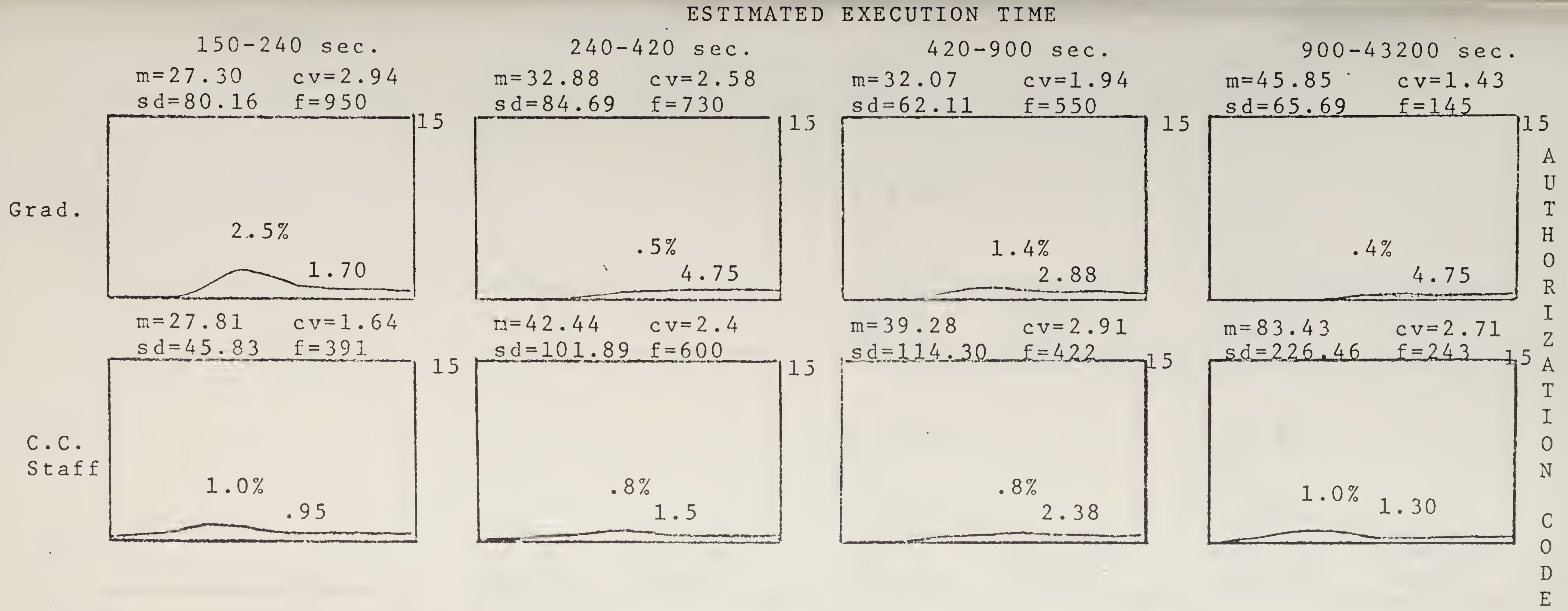
FIGURE 22.



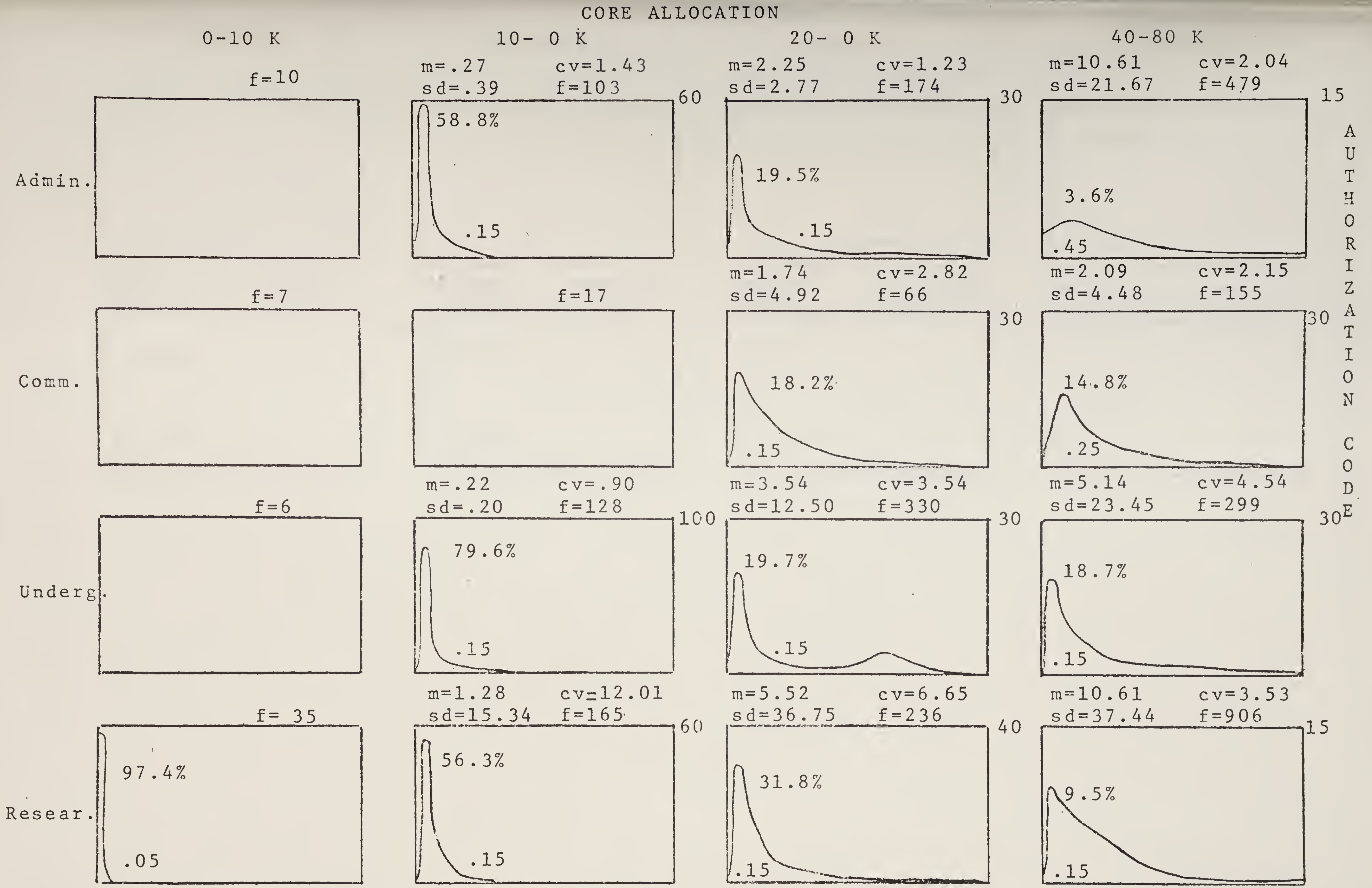
TRT CONDITIONED ON ESTIMATED EXECUTION TIME AND AUTHORIZATION CODE



TRT CONDITIONED ON ESTIMATED EXECUTION TIME AND AUTHORIZATION CODE



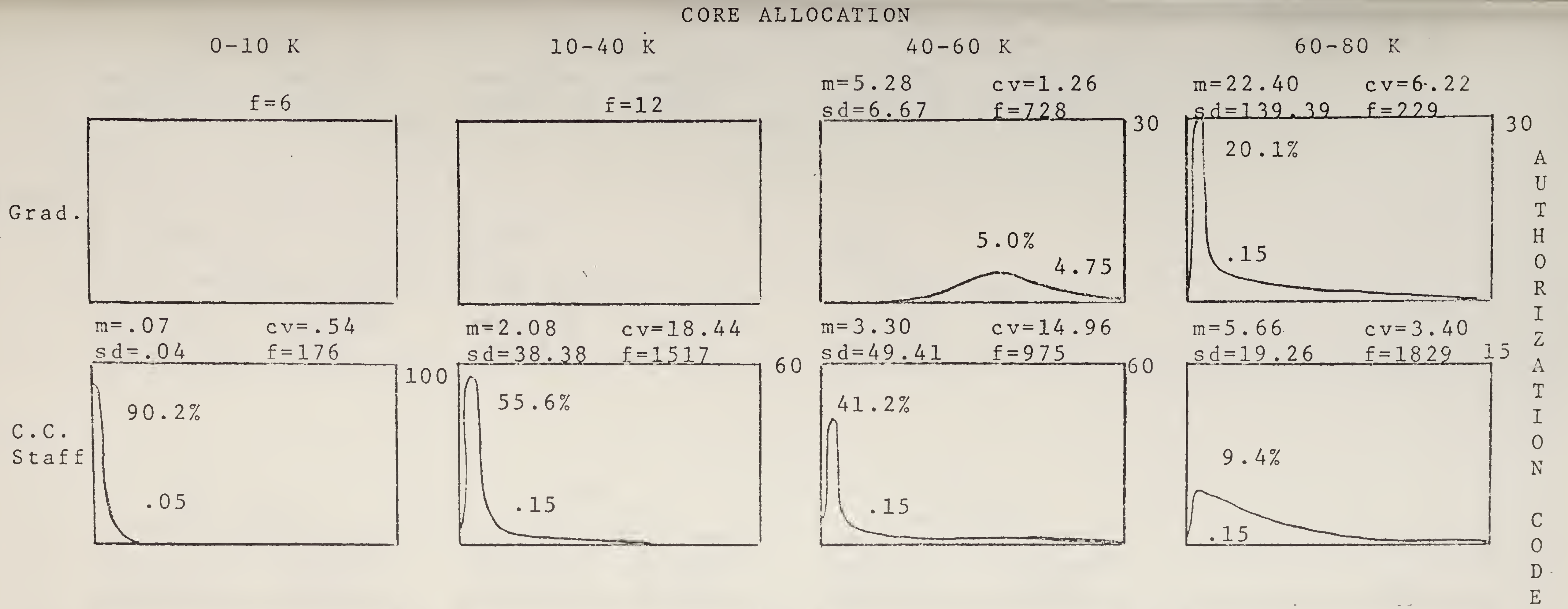
TRT CONDITIONED ON ESTIMATED EXECUTION TIME AND AUTHORIZATION CODE



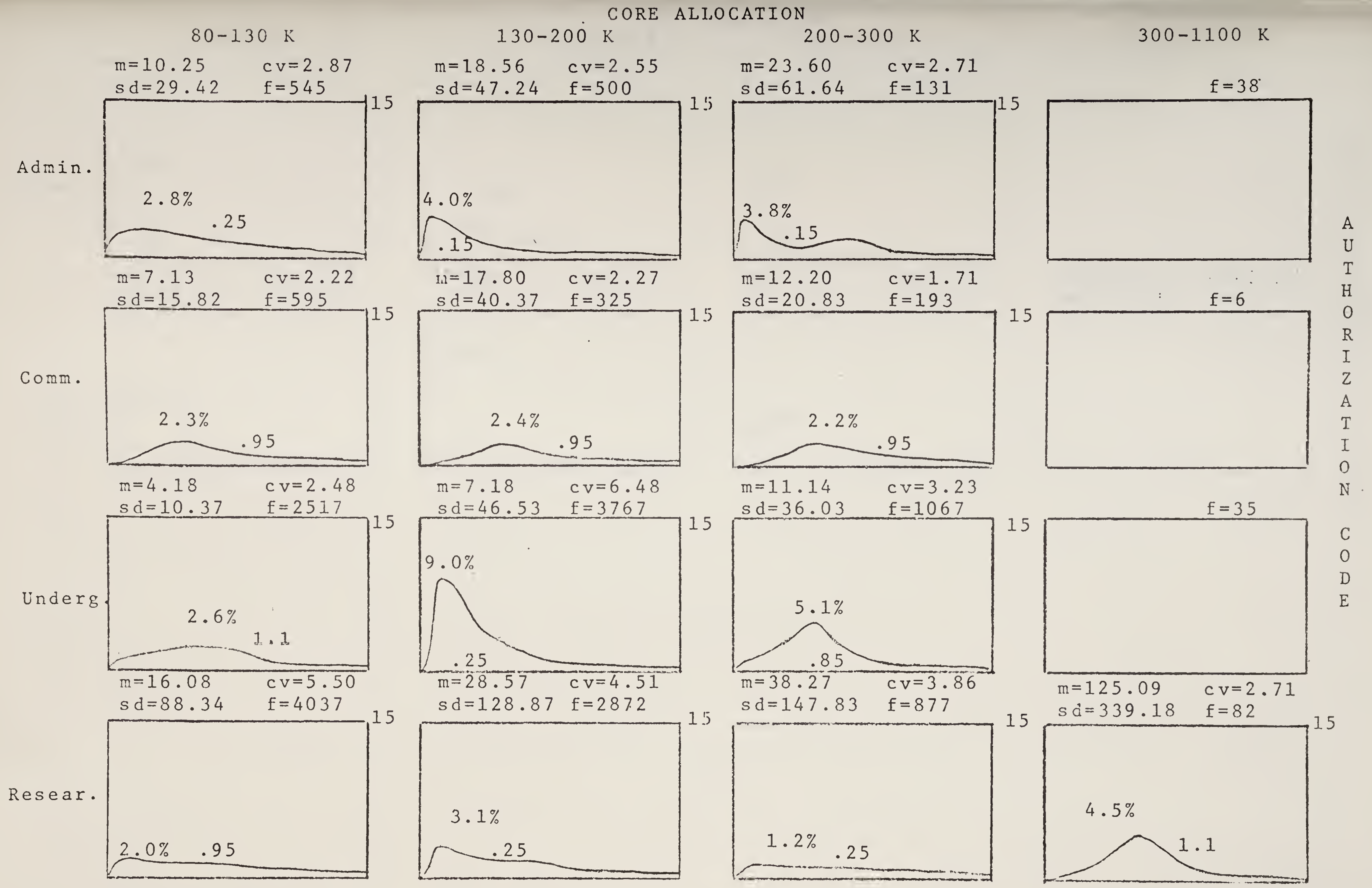
CPUT CONDITIONED ON CORE ALLOCATION AND AUTHORIZATION CODE

FIGURE 23.

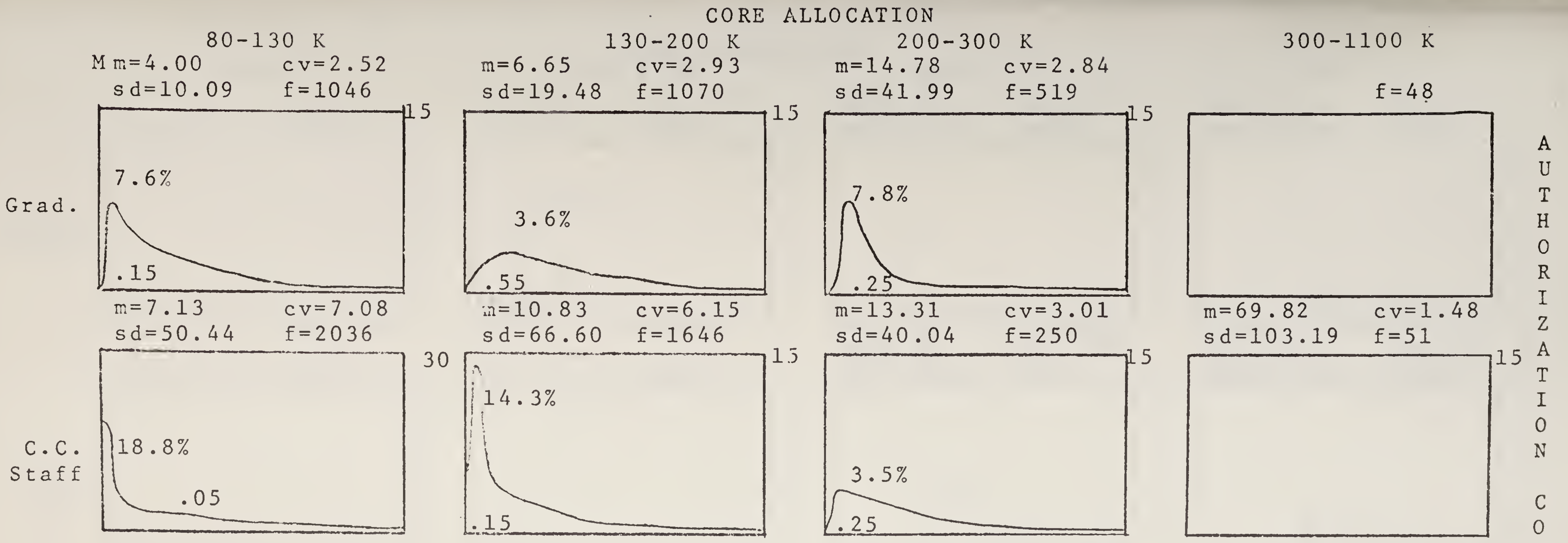
Page 1 of 4



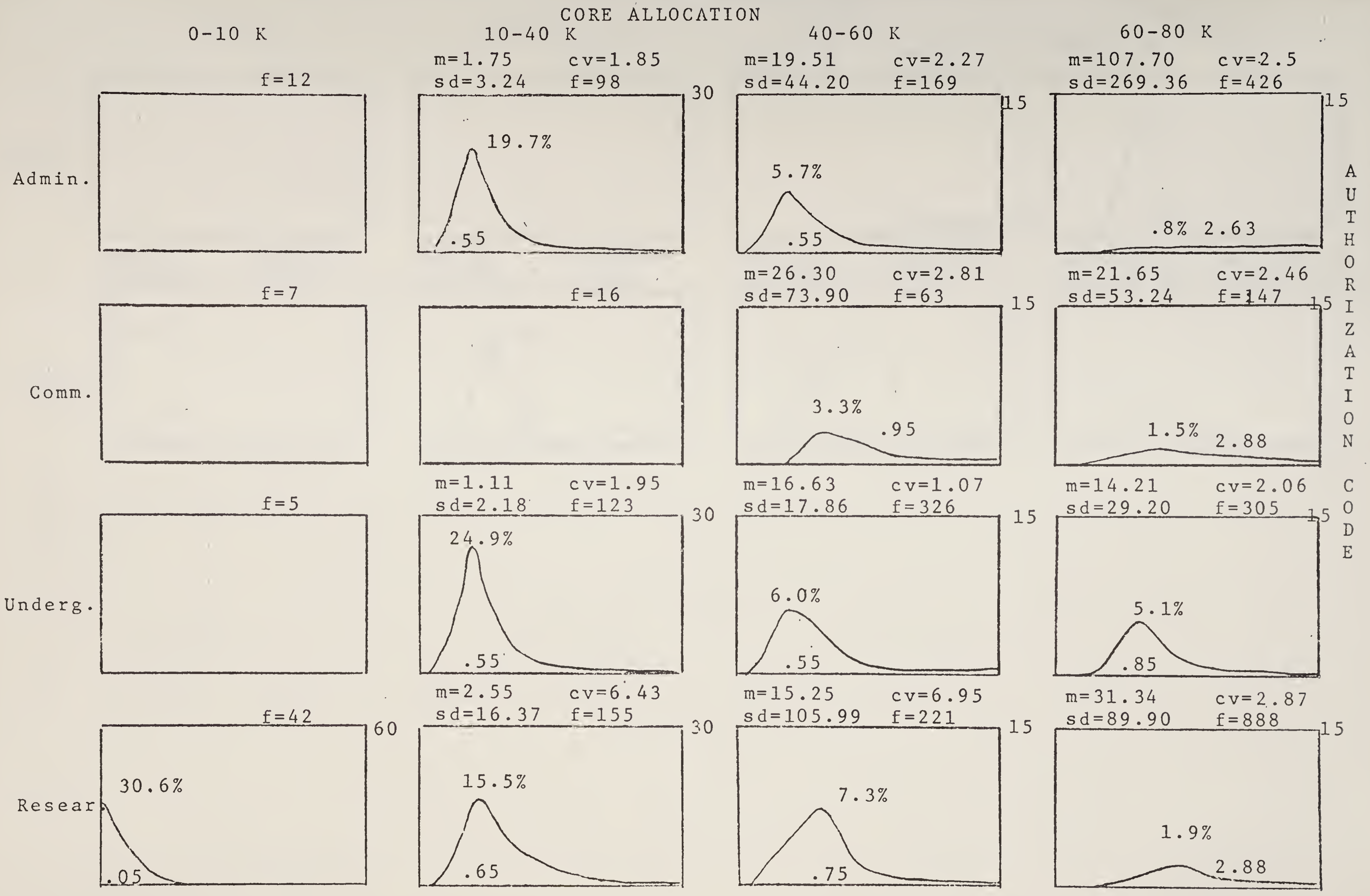
CPUT CONDITIONED ON CORE ALLOCATION AND AUTHORIZATION CODE



CPUT CONDITIONED ON CORE ALLOCATION AND AUTHORIZATICN CODE

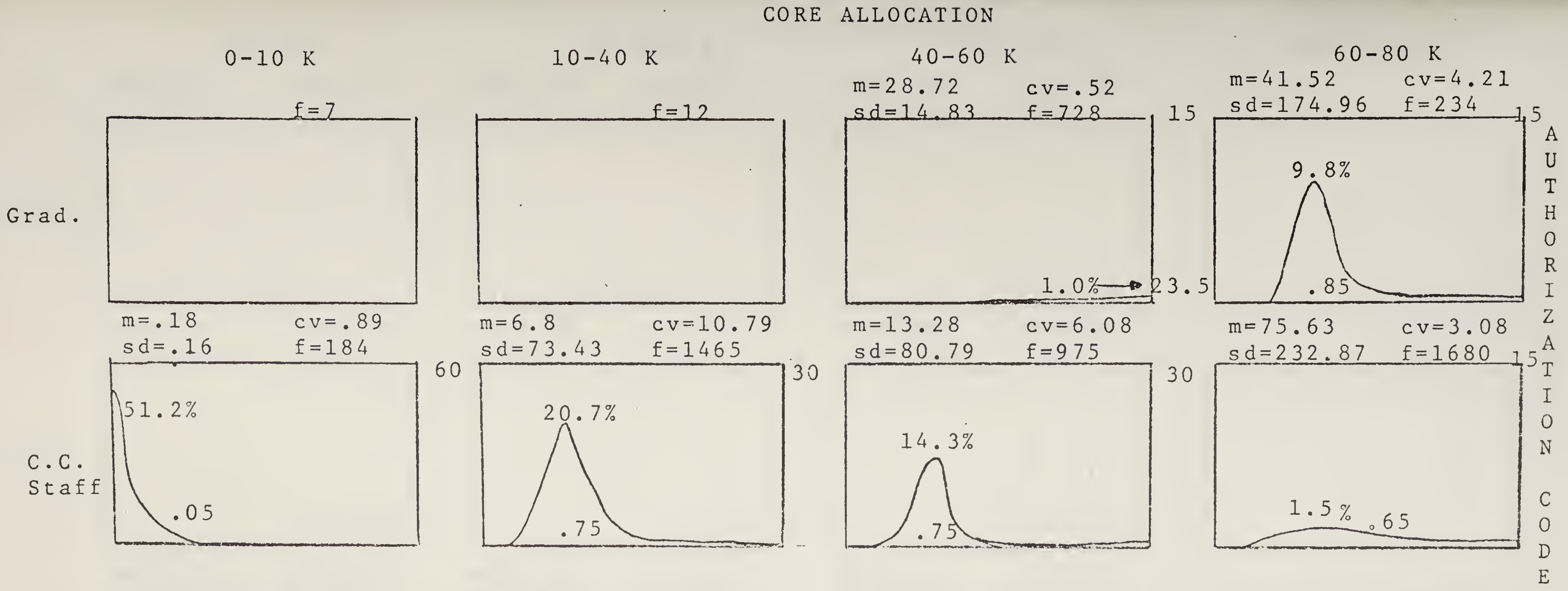


CPUT CONDITIONED ON CORE ALLOCATION AND AUTHORIZATION CODE

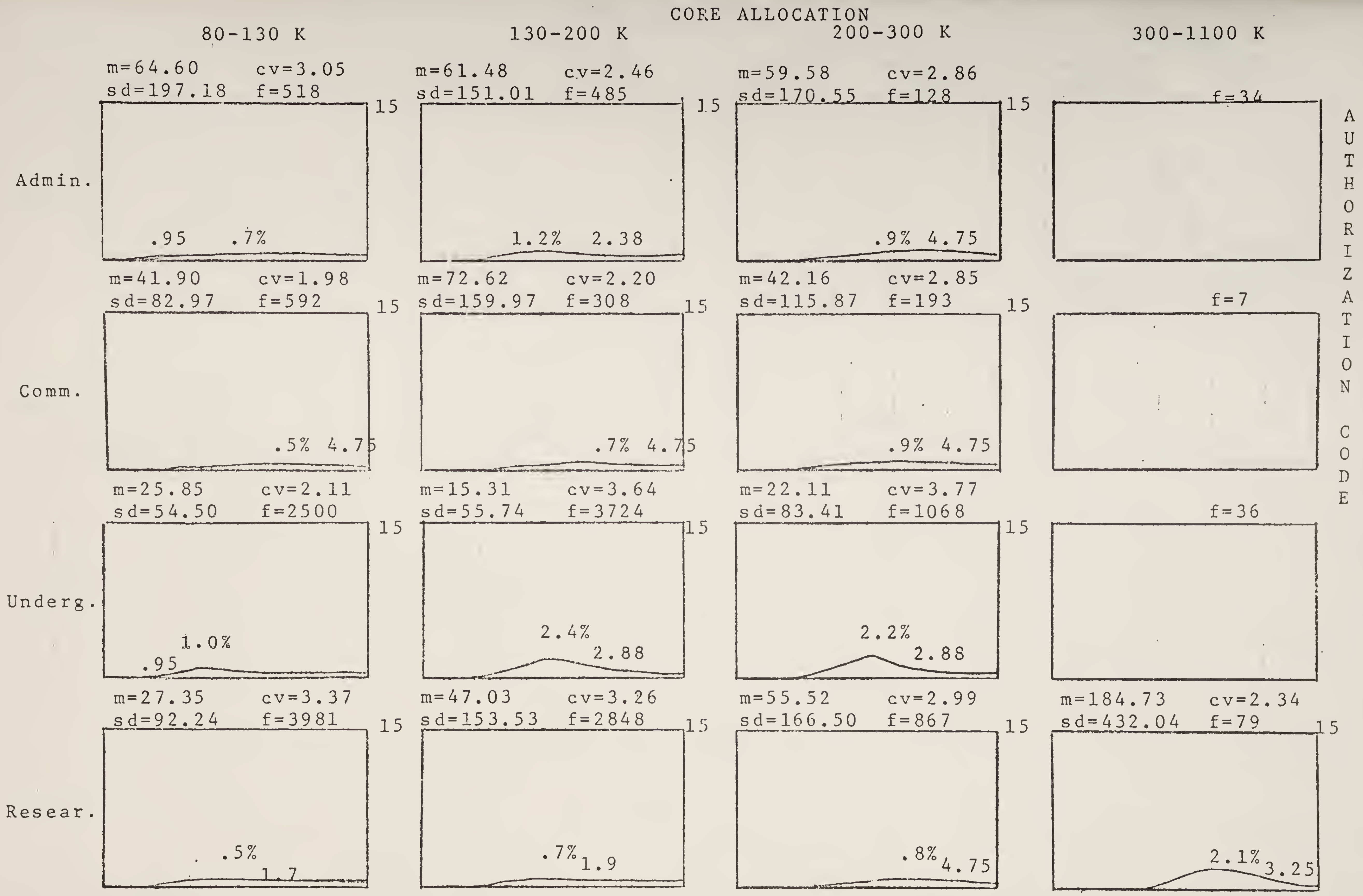


TRT CONDITIONED ON CORE ALLOCATION AND AUTHORIZATION CODE
FIGURE 24.

Page 1 of 4

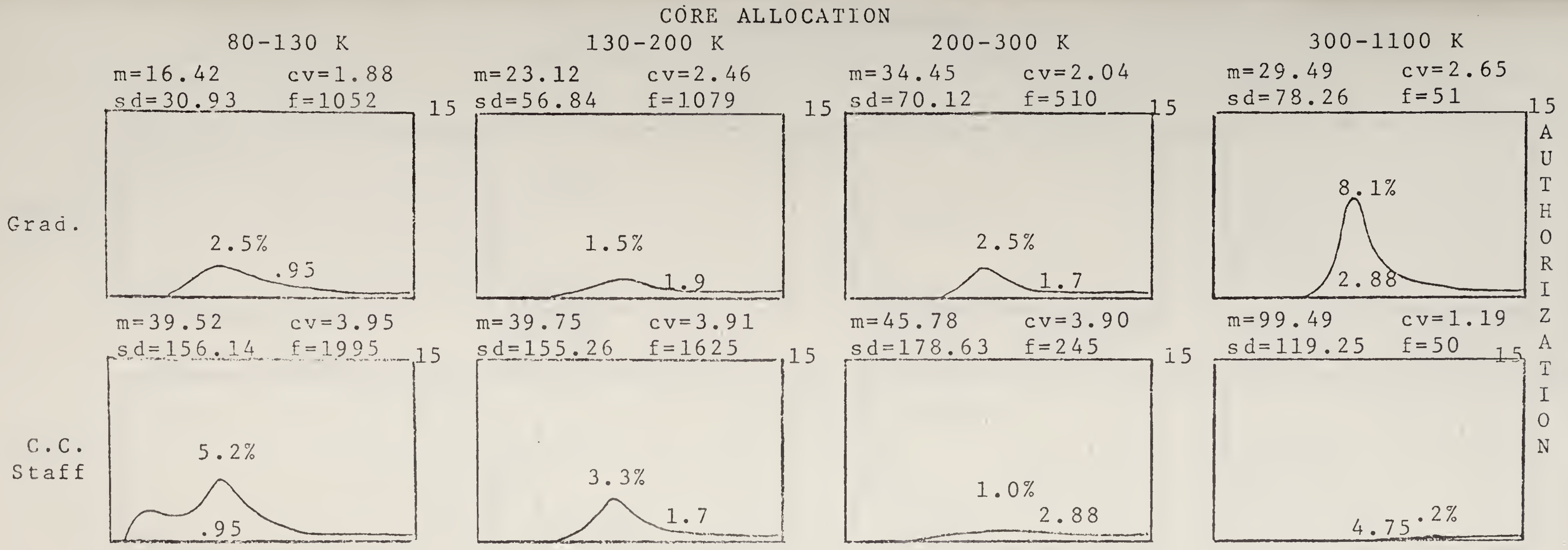


TRT CONDITIONED ON CORE ALLOCATION AND AUTHORIZATION CODE

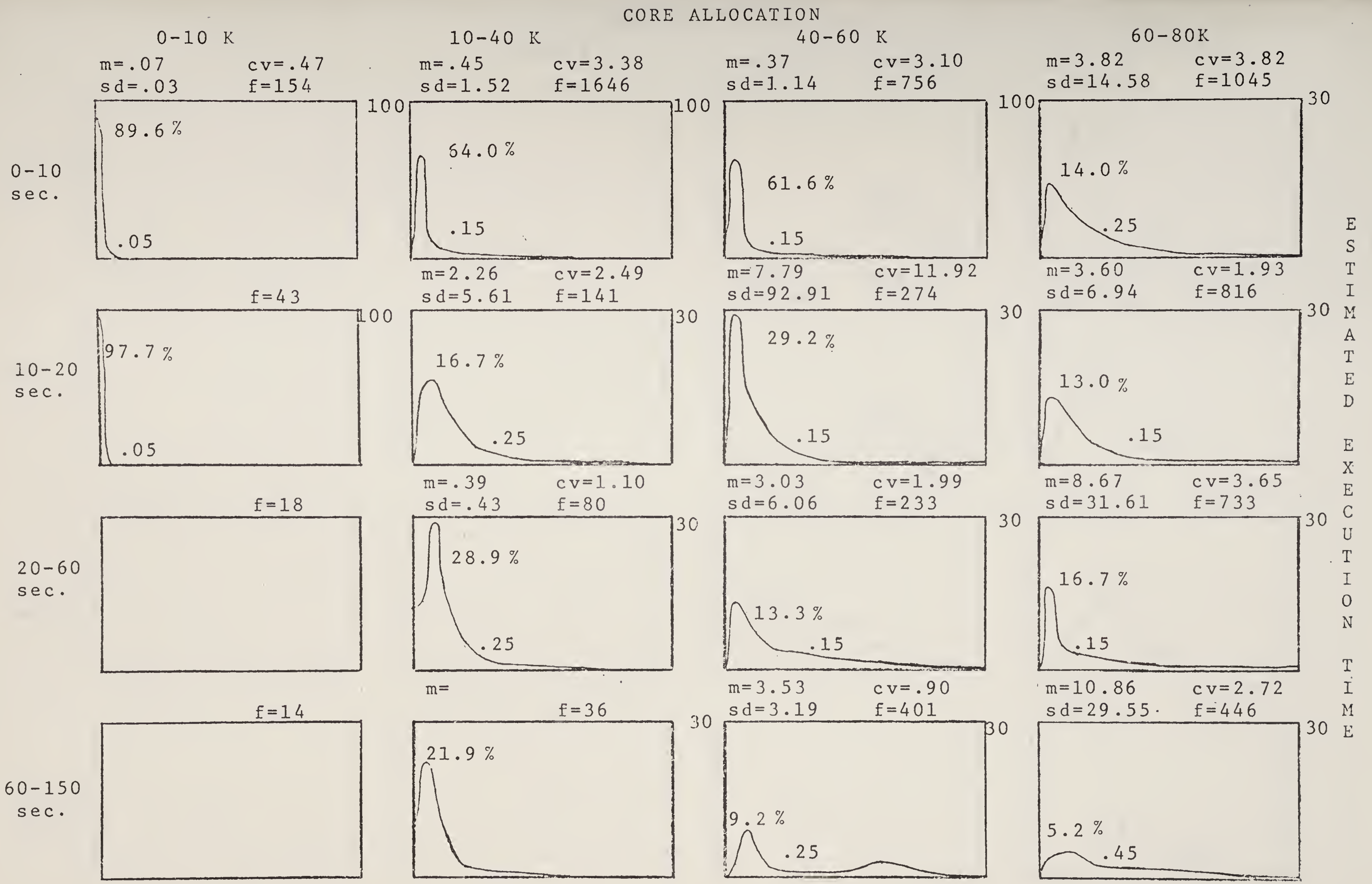


TRT CONDITIONED ON CORE ALLOCATION AND AUTHORIZATION CODE

Page 3 of 4



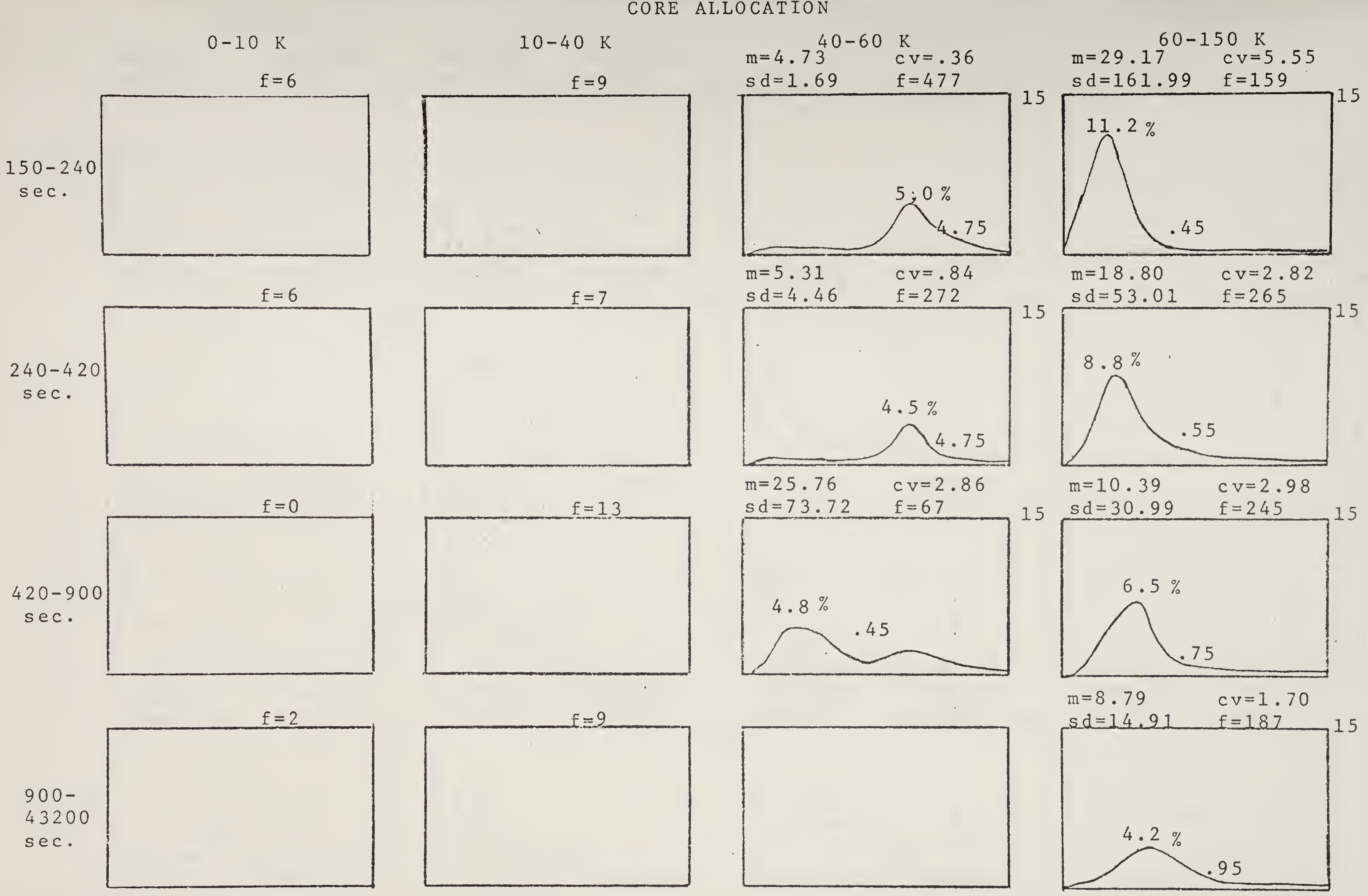
TRT CONDITIONED ON CORE ALLOCATION AND AUTHORIZATION CODE



CPUT CONDITIONED ON ESTIMATED EXECUTION TIME AND CORE ALLOCATION

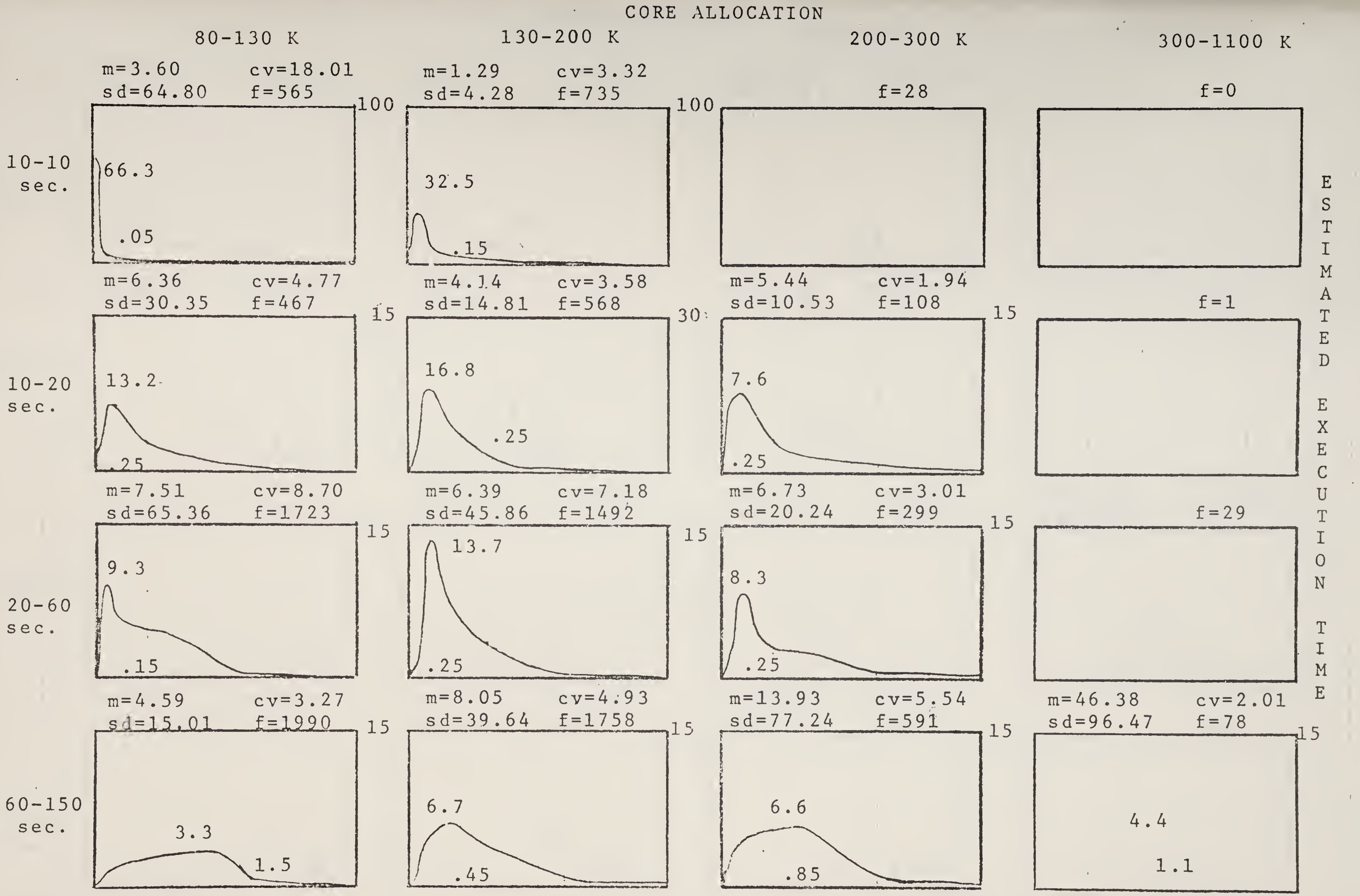
FIGURE 25.

Page 1 of 4

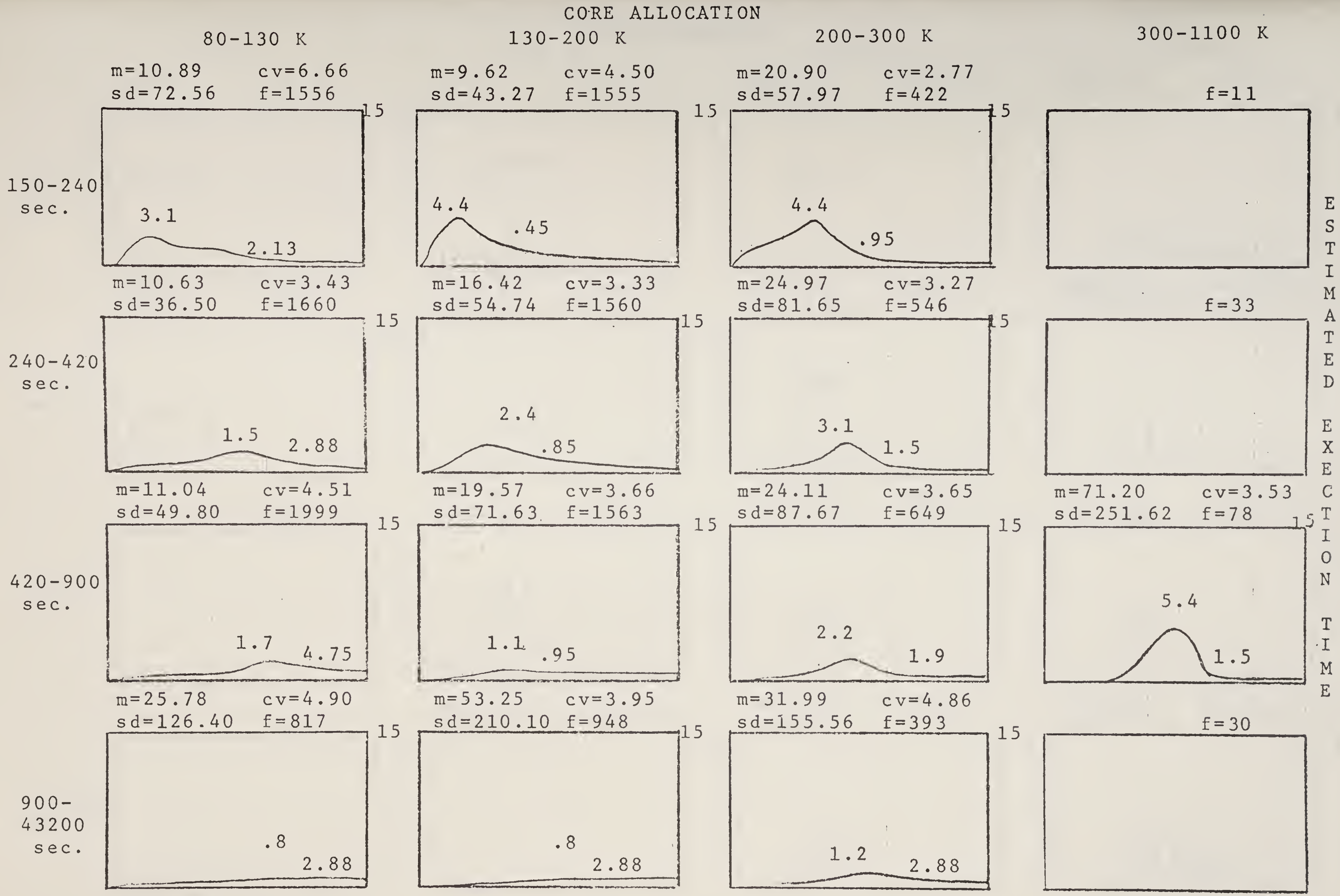


CPUT CONDITIONED ON ESTIMATED EXECUTION TIME AND CORE ALLOCATION

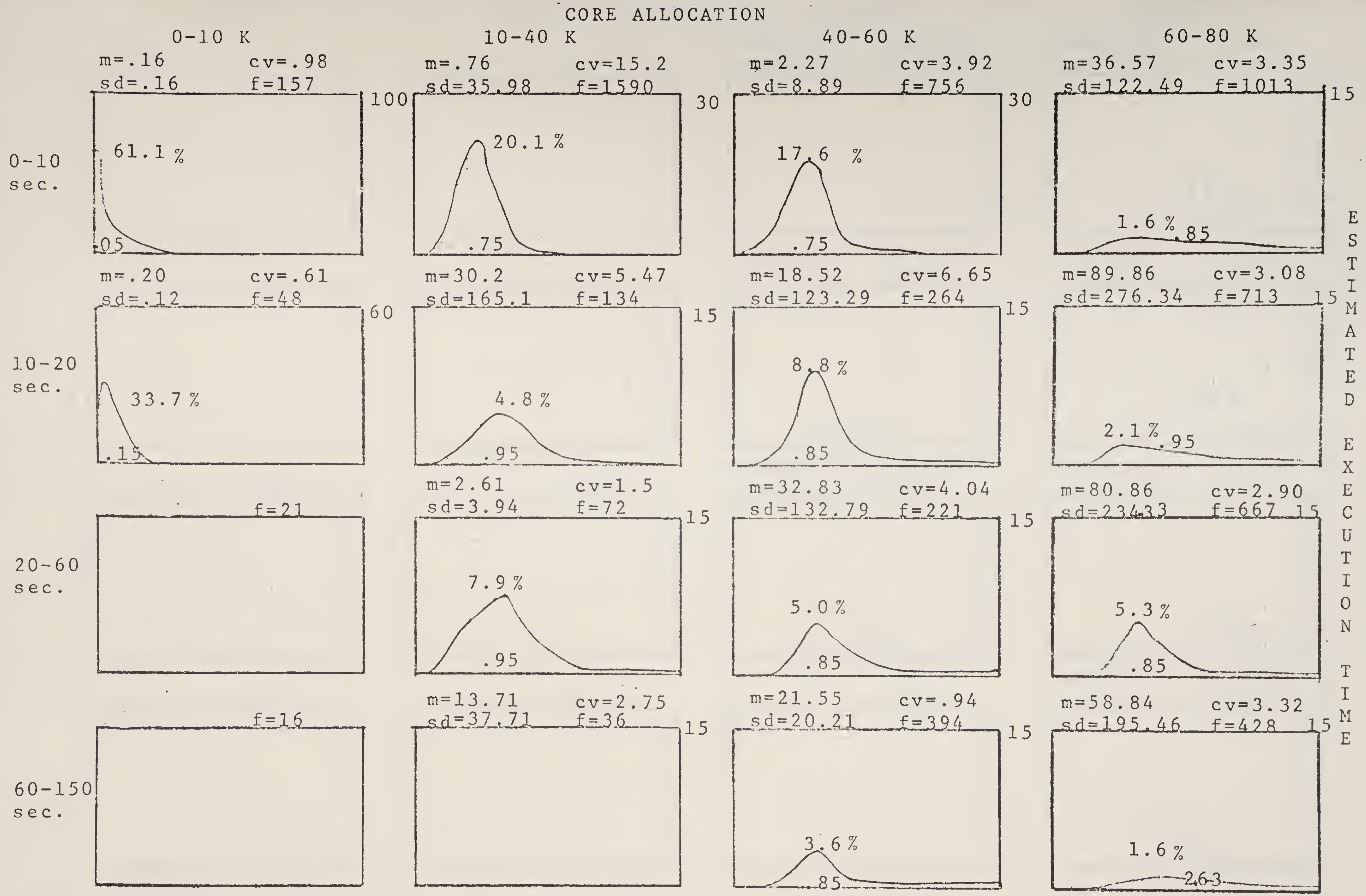
Page 2 of 4



CPUT CONDITIONED ON ESTIMATED EXECUTION AND CORE ALLOCATION

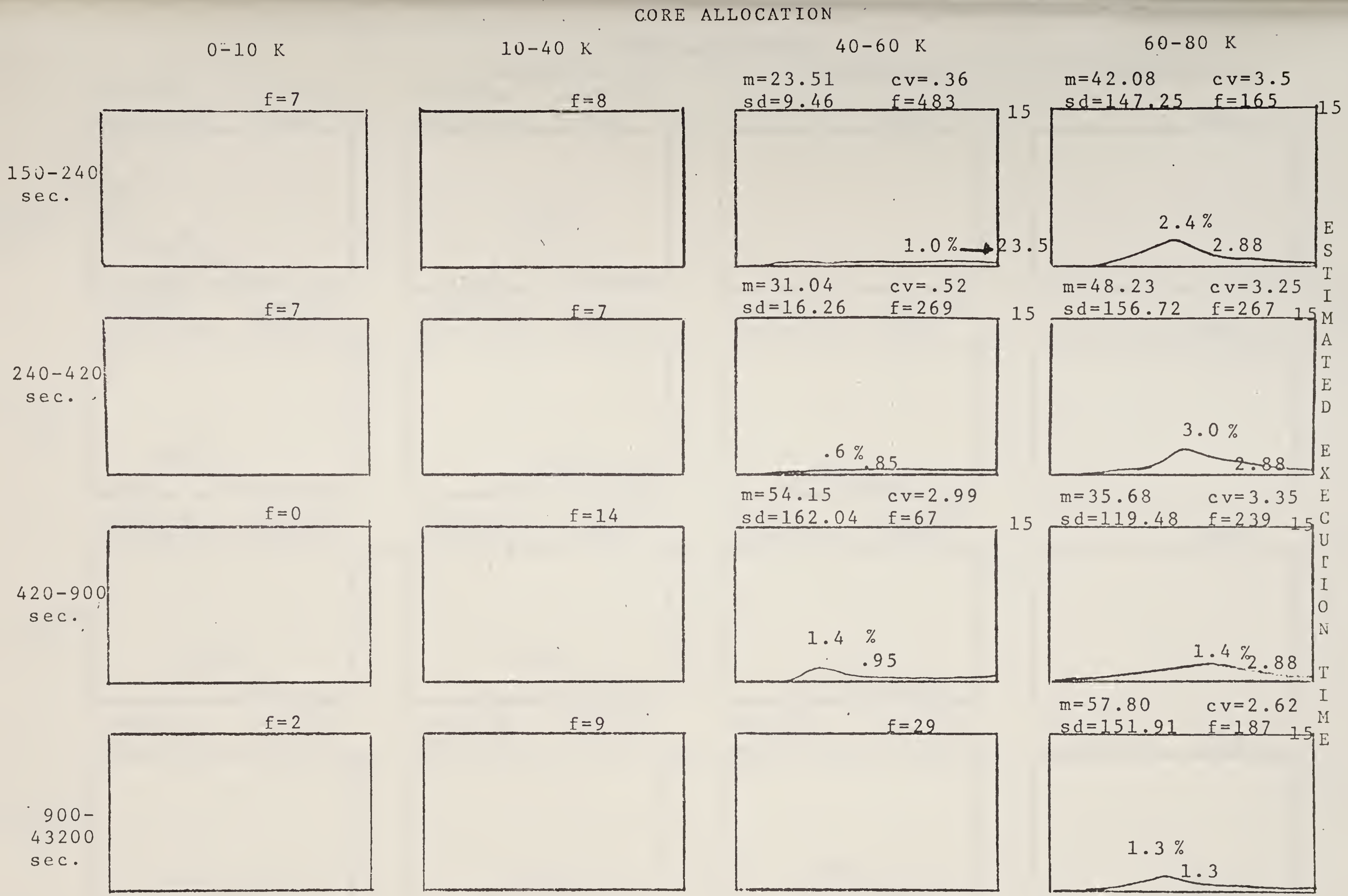


CPUT CONDITIONED ON ESTIMATED EXECUTION TIME AND CORE ALLOCATION

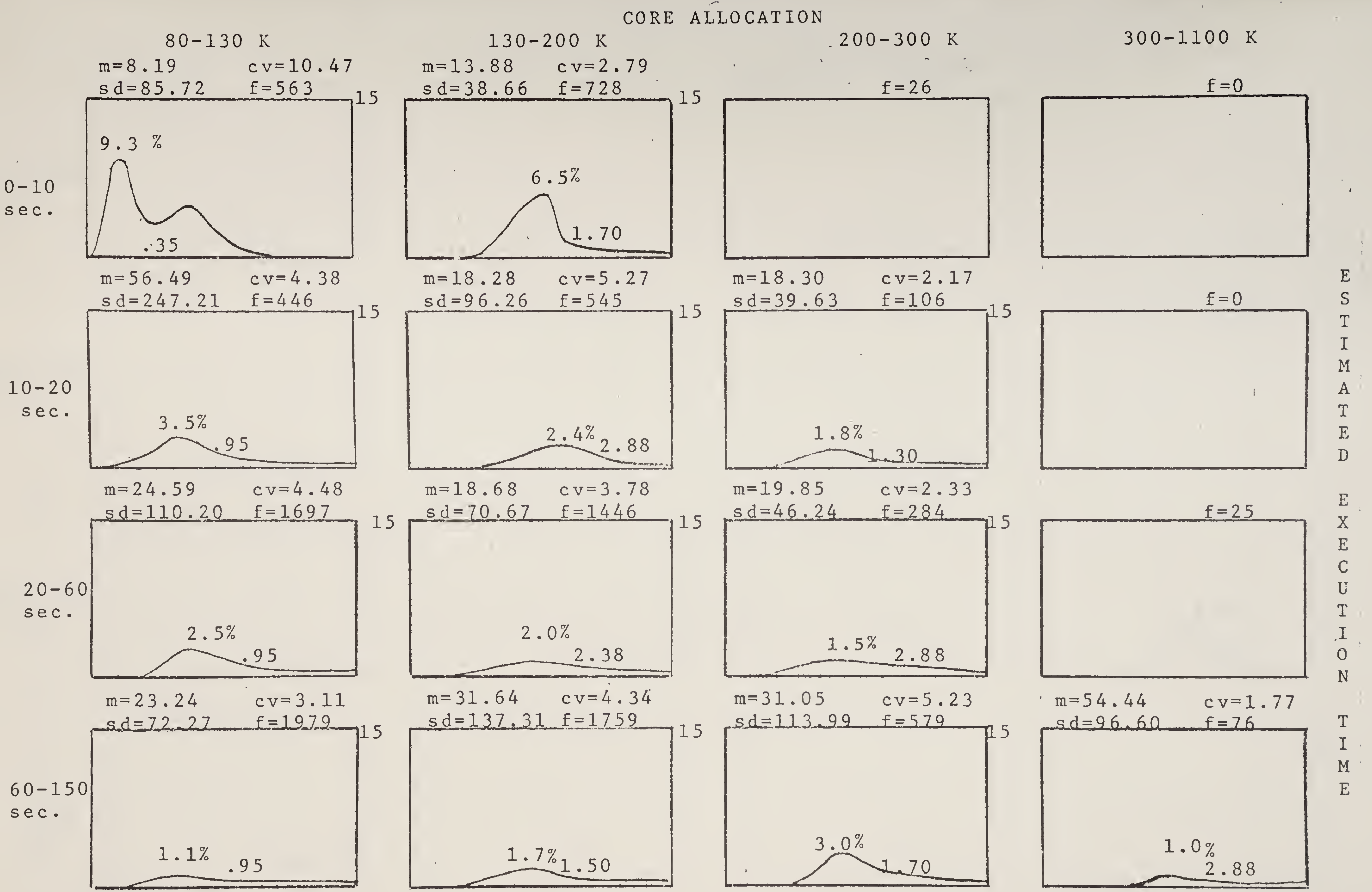


TRT CONDITIONED ON ESTIMATED EXECUTION TIME AND CORE ALLOCATION
FIGURE 26.

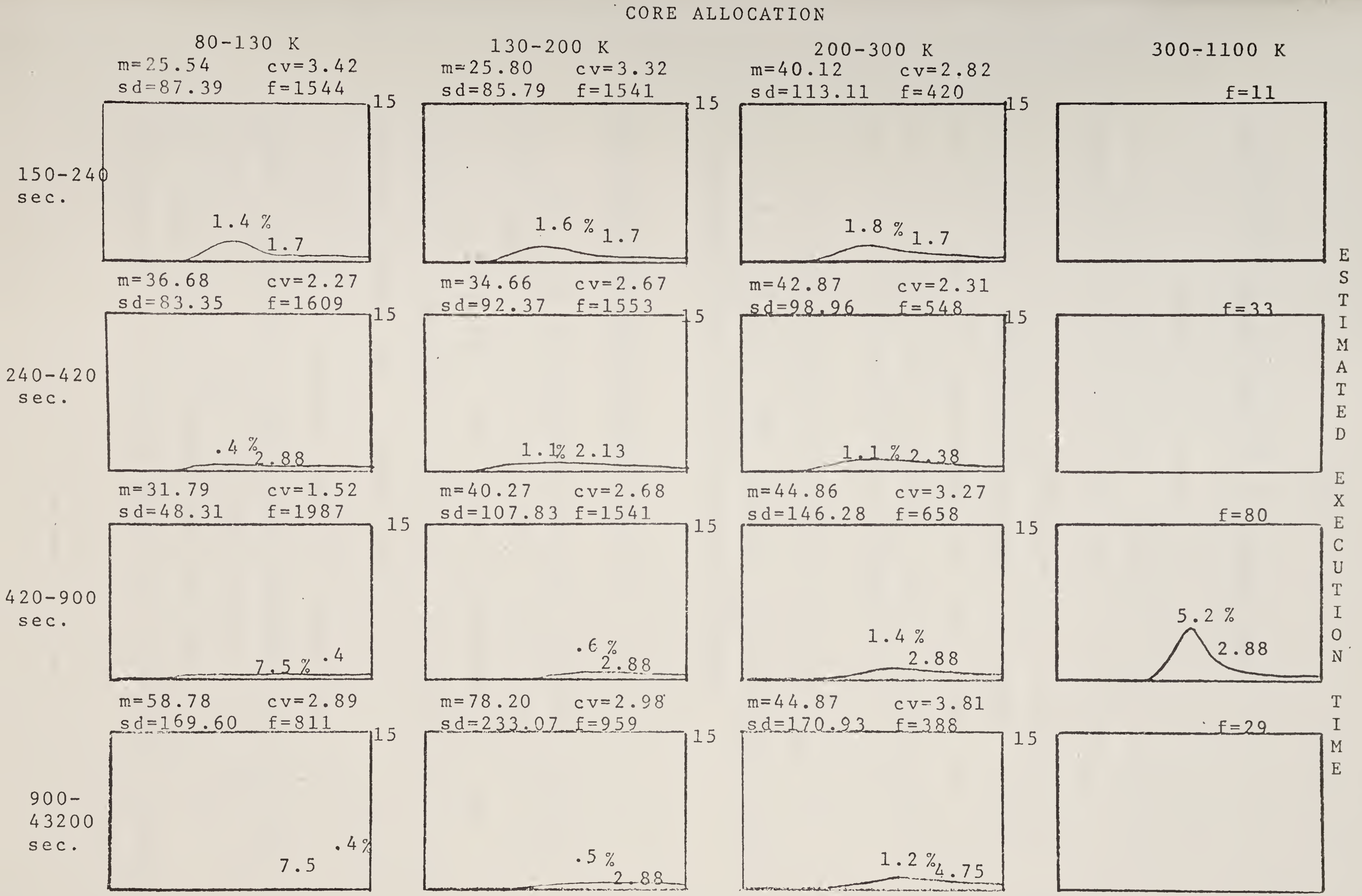
Page 1 of 4



TRT CONDITIONED ON ESTIMATED EXECUTION TIME AND CORE ALLOCATION



TRT CONDITIONED ON ESTIMATED EXECUTION TIME AND CORE



TRT CONDITIONED ON ESTIMATED EXECUTION TIME AND CORE ALLOCATION

Observations on the Empirical Distributions

Some interesting observations can be made concerning the empirical distributions. Generally, their shapes are gamma-like with extremely long tails. An occasional distribution resembles a hyperexponential or exponential distribution. The TRT distributions are similar to their corresponding CPUT distributions in that if one peaks, the other does also, but the peaks of the TRT distributions are much flatter and are displaced linearly to the right.

Particular observations of the distributions are provided below, along with insights where relevant, provided by knowledge of the IBM/370 operating system.

Zero Parameter Conditioning (Figure 6)

(1) The TRT distribution shows a greater frequency in the first 0 to .1 second interval than in the following one or two intervals. This may be attributable to jobs completing a compilation phase only, i.e., jobs that do not execute, probably due to syntax errors.

(2) As is expected, the CPUT distribution has a greater proportion of jobs in the beginning intervals than TRT does (0 to 1 seconds accounts for 35% of the jobs in the CPUT distribution, and only 11% of the jobs in the TRT distribution). It also has fewer jobs in the tail intervals

(10 to 2500 seconds accounts for 16% of the jobs in the CPUT distribution and 43% of the jobs in the TRT distribution).

(3) The coefficients of variation of the CPUT and TRT distributions are 5.9 and 3.7, respectively.

One Parameter Conditioning

(1) Class (Figures 7 and 8)

(a) CPUT and TRT distributions are distinct to some degree, but generally do not display much variation, although TRT distributions are better in this regard than CPUT distributions. Class A jobs of the TRT distributions displayed the characteristic described in Zero Parameter (1). The explanation there is supported by the reasoning that Class B and C jobs probably debug syntax errors in Class A since it provides faster turnaround time and is cheaper.

(b) Class M has the most distinctive distribution. Since this is the Computing Centre staff class, many short programs related directly to the function of the Centre are run, and many of these jobs run repeatedly throughout a day.

(c) All classes except M displayed long tails. TRT tails were more full than CPUT tails.

(d) The jobs were not distributed evenly over the classes. Class A had more than 75% of the jobs. Class B, C and M had approximately 10%, 5%, and 10%, respectively.

(e) The coefficients of variation were significantly less than the coefficients of variation for the unconditioned distributions in 75% of all the classes.

(2) Authorization Code (Figures 9 and 10)

(a) CPUT distributions are distinct but do not display much variation among classes, with the exception of the Computing Centre Staff class. Again, TRT distributions are more dissimilar from each other than the CPU distributions, reflecting the varying I/O demands of the different groups of users.

(b) The number of jobs in each of the Undergraduate, Research, and Computing Centre Staff classes is approximately the same and in all account for more than 75% of the jobs. The Graduate, Administration, and Commercial classes have close to 11%, 5.5%, and 3.5%, respectively.

(c) The coefficients of variation for the CPUT and TRT distributions are significantly less than those for the unconditioned distributions in 67% and 83% of the classes, respectively.

(3) Estimated Execution Time (Figures 11 and 12)

(a) Both CPUT and TRT distributions show distinct distributions in all eight classes. Both show a good degree of variation in the shapes of the distributions.

(b) As is expected, there is positive correlation between increasing processing times and increasing estimated execution times. For example, the means and maximums of the CPUT distributions rise from 1.[illegible] seconds and .16 seconds, respectively, for the 0-10 second estimated execution time interval, to 35.85 seconds and 3.[illegible] seconds, respectively, for a 900-432000 second interval. Those of the TRT distributions rise from 12.3 seconds and .76 seconds, respectively, for a 0-10 second interval to 63.24 seconds and 4.25 seconds, respectively, for a 900-43200 second interval.

(c) The classes have relatively equal numbers of jobs in each.

(d) A peculiarity of this system can be seen: when the estimated execution time is equal to actual CPU time plus I/O time, the job should be terminated. In theory, there should be no TRT value greater than the upper limit of the estimated execution time interval. However, the system does not always check for this set of events.

(e) Only in 50% of the classes of both CPUT and TRT is the coefficient of variation less than that of the relevant unconditioned distribution. The decreasing coefficients of variation with increasing estimated execution times, imply that, for longer jobs, this parameter is a better indicator of service times.

(4) Core Allocation (Figures 13 and 14)

(a) As in the Estimated Execution Time results, CPUT and TRT distributions conditioned on Core Allocation are distinct and show a good degree of variation.

(b) The means, the rightward movement of the peaks, and the proportions in the tails generally demonstrate positive correlation with increasing amounts of core allocation. Means and maximums for CPUT distributions rise from .06 and .05 seconds, respectively, for a 0-10K core interval to .66 and 1.3 seconds, respectively, for a 300-1100K interval. Those for TRT distributions rise from .18 and .05 seconds, respectively, for a 0-10K interval to 97.8 and 2.9 seconds, respectively, for a 300-1100K interval. However, the relationship for the core allocation parameter is not as consistent in each of the core allocation intervals, as it is for the estimated execution time parameter in each of its intervals. But it is also more consistent in TRT distributions than in CPUT distributions. This may be due to the effect of the operating system on core allocation, i.e., the default core allocation assigned to a job when none is specified by the user. The effect would be lessened in TRT distributions because of the dispersion caused by additional I/O time.

(c) The jobs are not distributed evenly over the core allocation classes. Intervals of 80-130K and 130-200K each have close to 33% of the jobs. 0-10K and 300-1100K each have only 1% of the jobs. The remainder (32%) is distributed relatively evenly over the four remaining intervals.

(d) The 0-10K interval displays a very high-peaked, short-tailed, non-increasing distribution. The 80-130K interval for CPUT also displays a non-increasing distribution, but with a long tail. There are no known reasons for these anomalies.

(e) In less than 50% of the classes is the coefficient of variation significantly less than those of the unconditioned distributions.

Two Parameter Conditioning

For ease of discussion, particular distributions will be referred to as CPUT/P1/P2 or TRT/P1/P2 where P1 and P2 are either conditioning parameters or specific classes of the conditioning parameters.

(1) Class and Estimated Execution Time (Figures 15 and 16)

(a) Distributions are distinct to some degree for most CPUT classes. TRT distributions are less distinct--the following intervals could be combined: TRT/B/150-240sec. and TRT/B/150-240sec.; TRT/A/420-900sec. and TRT/A/1900-43200sec.; TRT/C/240-420sec. and TRT/C/420-900sec. The sets of distributions in Classes A and B generally do not show much variation from each other for either of CPUT or TRT. Class C distributions have generally flatter peaks than those of Classes A and B, for CPUT; TRT distributions are all very flat. Class M distributions are very distinct from those of other classes.

(b) The relationship between Estimated Execution Time and both of CPUT and TRT, evident in one parameter conditioning is also evident in this two parameter conditioning.

(c) The percentage of jobs in each set of classes for the Class parameter is distributed fairly evenly over the eight execution time classes, with the exception of Class M. Here, the first interval, 0-10 seconds, accounts for 92% of the jobs in Class M and virtually no jobs have estimated execution times of greater than 150 seconds.

(d) In greater than 80% of the classes for both CPUT and TRT, the coefficient of variation is significantly less than the coefficient of variation of the relevant unconditioned distribution.

(e) 6 out of the 32 possible classes in both CPUT and TRT conditioning contain an insufficient number of jobs.

(2) Class and Core Allocation (Figures 17 and 18)

(a) The distributions are distinct for all CPUT classes. TRT distributions have some similarities--TRT/B/130-200K and TRT/B/200-300K; TRT/C/200-300K and TRT/C/300-1100K can be combined. Unlike CPUT/Class/Estimated Execution Time, the sets of CPUT distributions for Classes A, B, C and M all show significant variation from each other. TRT distributions show significant variation in lower core classes but classes involving core allocation greater than 130K were quite similar for Classes A, B, and C.

(b) The relationship between core allocation and CPUT and TRT is evident here in each set of distributions for Classes A, B, C, and M.

(c) As is expected from relevant observations in one parameter conditioning, the percentage of jobs is distributed unevenly over all of the classes.

(d) Greater than 85% of the classes for both CPUT and TRT have coefficients of variation significantly less that that of the relevant unconditioned distribution.

(e) 6 out of the 32 possible classes in both CPUT and TRT conditioning contain an insufficient number of jobs.

(f) The non-increasing distribution of the core allocation class 0-10K, noted previously, recurs in all classes of CPUT/Class/0-10K, but not in every class of TRT/Class/0-10K because of the effect of I/O time. The non-increasing distribution of a core allocation class of 80-130K, appears for CPUT/C/80-130K and CPUT/M/80-130K. However, Class M is primarily responsible for the anomaly since it contains 93% of the jobs causing the anomaly.

(3) Class and Authorization Code (Figures 19 and 20)

(a) CPUT distributions are distinct for all classes, but the shapes for Classes A, B, and C are generally more similar to each other that are those of CPUT/Class/Estimated Execution Time or CPUT/Class/Core allocation. TRT distributions are distinct to some degree, but the distributions for Classes A, B, and C are all very flat and vary much less than the corresponding CPUT distributions and TRT/Class/Estimated Execution Time or TRT/Class/Core Allocation. Class M distributions are quite similar to each other for 3 of the 4 classes represented. This is an expected characteristic of Class M, as it is reserved for emergency jobs usually involving some repair of the system.

(b) As is expected from previous observations, the distribution of the percentage of jobs in each class is poor.

(c) Only 2 classes out of the possible 24 for both CPUT and TRT have an insufficient number of jobs.

(d) Greater than 70% and 90% of the coefficients of variation of the CPUT and TRT distributions, respectively, are significantly less than that of the relevant unconditioned distribution.

(4) Authorization Code and Estimated Execution Time (Figures 21 and 22)

(a) All distributions of CPUT are distinct but do not vary in the height of the peaks as greatly as CPUT/Estimated Execution Time/Core Allocation. The distributions of TRT are somewhat distinct--TRT/Undergraduate/420-900sec. and TRT/Research/420-900sec. can be combined--but are usually very flat for any estimated execution time greater than 60 seconds.

(b) The relationship between CPUT and Estimated Execution Time as discussed in (3)(b) of One Parameter

Conditioning, is again evident here for all classes of Authorization Code, but that between TRT and Estimated Execution Time is less so.

(c) The distribution of the percentage of jobs in each class is somewhat uneven.

(d) Greater than 75% of the CPUT and TRT distributions had coefficients of variation less than that of the corresponding unconditioned distribution.

(e) 47 out of the possible 48 classes are of sufficient frequency to be considered.

(5) Authorization Code and Core Allocation (Figures 23 and 24)

(a) The CPUT distributions are all distinct and show more variation in the shapes than do those for CPUT/Authorization Code/Estimated Execution Time, but less than those for CPUT/Core Allocation/Estimated Execution Time. Most distributions for TRT are distinct--TRT/Administration/130-200sec. and TRT/Commercial/130-200 can be combined--and the same can be said for variation as is said for CPUT distributions.

(b) The relationships between CPUT, TRT and Core Allocation mentioned in (4)(b) of One Parameter Conditioning, are evident here again.

(c) The percentage of jobs in each class is quite uneven.

(d) Greater than 65% of the classes for both CPUT and TRT have coefficients of variation greater than that of the relevant unconditioned distribution.

(e) 10 of the possible 48 classes for both CPUT and TRT contain an insufficient number of jobs.

(6) Core Allocation and Estimated Execution Time (Figures 25 and 26)

(a) CPUT distributions are all distinct and vary greatly for all classes with frequency greater than 50. TRT distributions are less distinct, particularly for classes with an estimated execution time greater than 150 seconds-- TRT/200-300K/240-420sec. and TRT/130-200K/240-420sec.; TRT/200-300K/420-900sec. and TRT/200-300K/900-43200sec. can be combined. The distributions are usually very flat beyond this point.

(b) The relationships between CPUT, TRT, Core Allocation, and Estimated Execution Time discussed in (3)(b) and (4)(b) of One Parameter Conditioning, are evident here again.

(c) The classes have relatively unequal percentages of jobs.

(d) 80% and 60% of the CPUT and TRT distributions, respectively, have coefficients of variation less than that of the relevant unconditioned distribution.

(e) 18 out of the possible 64 classes for both CPUT and TRT contain an insufficient number of jobs.

Distribution Fitting

Further processing of the data involved attempts to fit the empirical distributions to theoretical distributions. Because of the previous work in the area of scheduling and service time distributions, and because of the general shape of the empirical distributions, the theoretical distributions considered were:

(1) exponential,
(2) hyperexponential,
(3) gamma,
(4) Weibull,
(5) log-normal,
(6) Beta, and
(7) convolution of other distributions.

We will discuss the treatment and results of each.

(1) Exponential

The exponential density distribution was produced in the main program described in Chapter II, along with the chi-square value.

Results

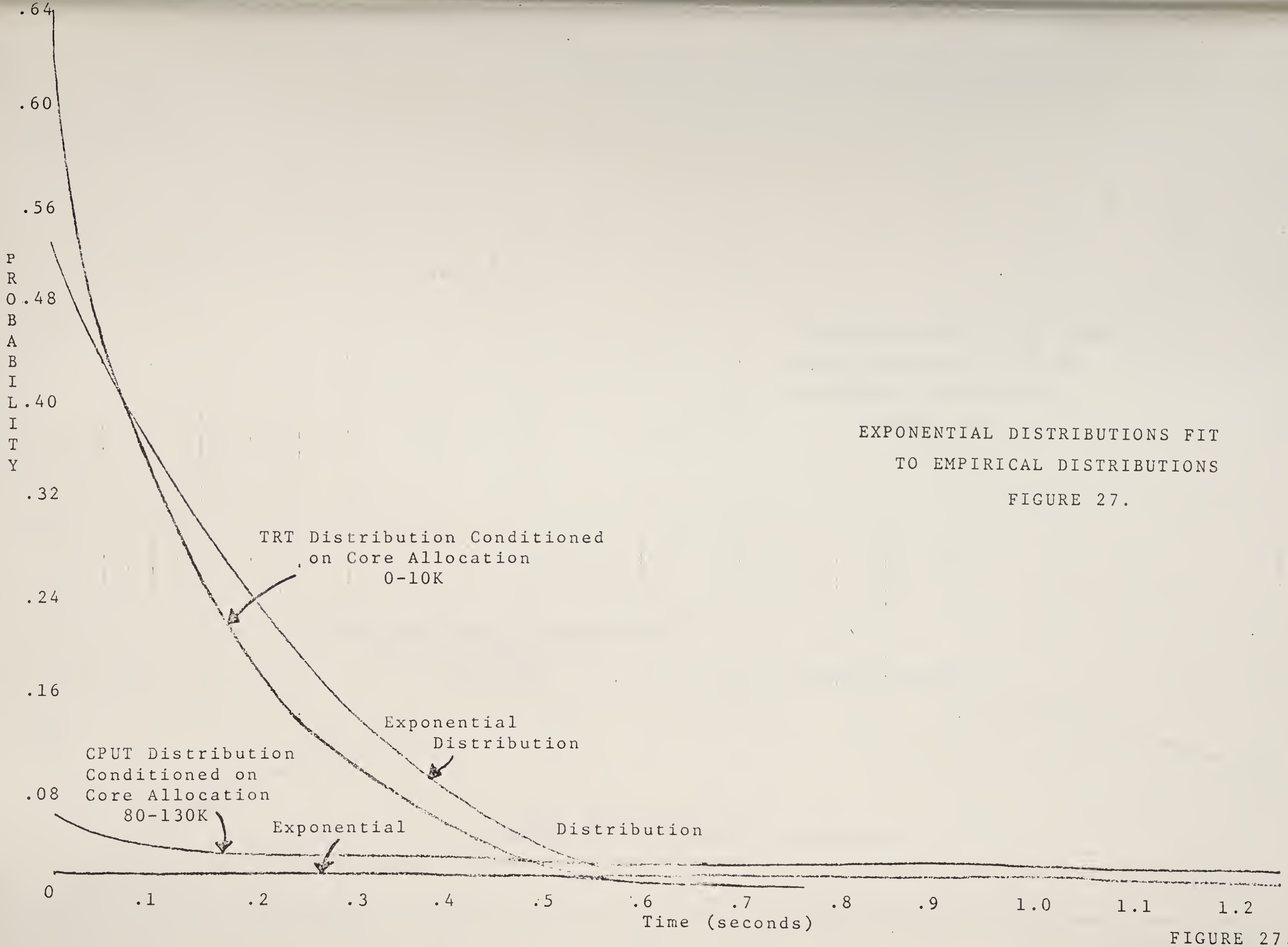
The exponential distribution is, generally, far from satisfactory. Only in very isolated cases, does the empirical distribution have an exponential-like shape. Figure 27 illustrates the difference between the empirical and the exponential distribution, in the best of those isolated cases.

(2) Hyperexponential and (3) Gamma

Here also, the theoretical density distributions were produced in the main program, with chi-square values. Tables reflecting cumulative probabilities for various values of the parameters were also produced.

Results

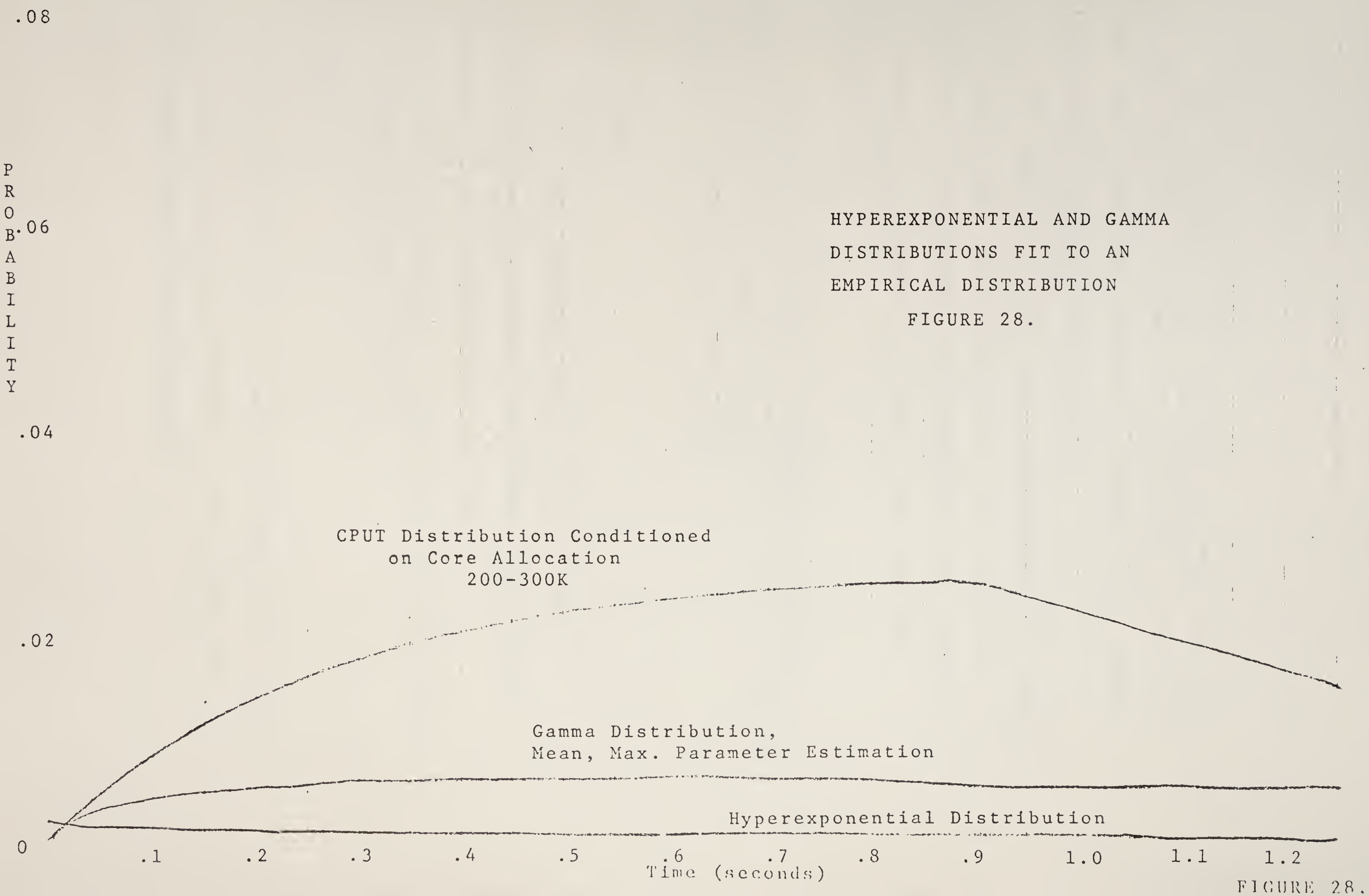
Again, the theoretical density distributions are very different from the empirical distributions. Figure 28 depicts this difference in one of the better fitting distributions. The tables do not prove to be any more satisfactory. Upon comparing the cumulative probabilities of the theoretical versus the empirical, we find that two conditions exist:



EXPONENTIAL DISTRIBUTIONS FIT
TO EMPIRICAL DISTRIBUTIONS

FIGURE 27.

FIGURE 27.



HYPEREXPONENTIAL AND GAMMA DISTRIBUTIONS FIT TO AN EMPIRICAL DISTRIBUTION

FIGURE 28.

FIGURE 28.

(1) Matching of the peaks produce very poor results in the tail area. In fact, the theoretical distributions lose their tails very quickly as compared to the extended tails of the empirical distributions.

(2) Matching of the tails produce very poor results in the peak area. The proportion of jobs in the peak of the theoretical distribution is insignificant when compared to the proportion of jobs in the peak of the empirical distribution.

Results for the gamma distribution are similar for estimation of its parameters by both the mean and standard deviation, and the mean and maximum.

(4) Weibull

In separate programs, tables of cumulative probabilities were generated for various values of the parameters of the distribution, and theoretical distributions were produced based on the regression method of estimating Weibull parameters.

Results

The fits here are very much like those of the gamma distribution. Through examination of the tables, and with a composite of Weibull distributions:

$$F_i(t) = 1 - e^{-(t/B_i)^{A_i}}, \quad a_i < t \le a_{i+1}, \quad i = 1,\ldots,m$$

$$\text{where } a_1 = 0 \text{ and } a_{m+1} = \infty,$$

a poor fit was obtained, which is the best thus far. Figure 29 gives an idea of the fit resulting from a composite of 5 distributions.

(5) Log-normal

Little was done with this distribution. This is an area for further investigation; the tail may be thicker than that of either the Weibull or gamma.

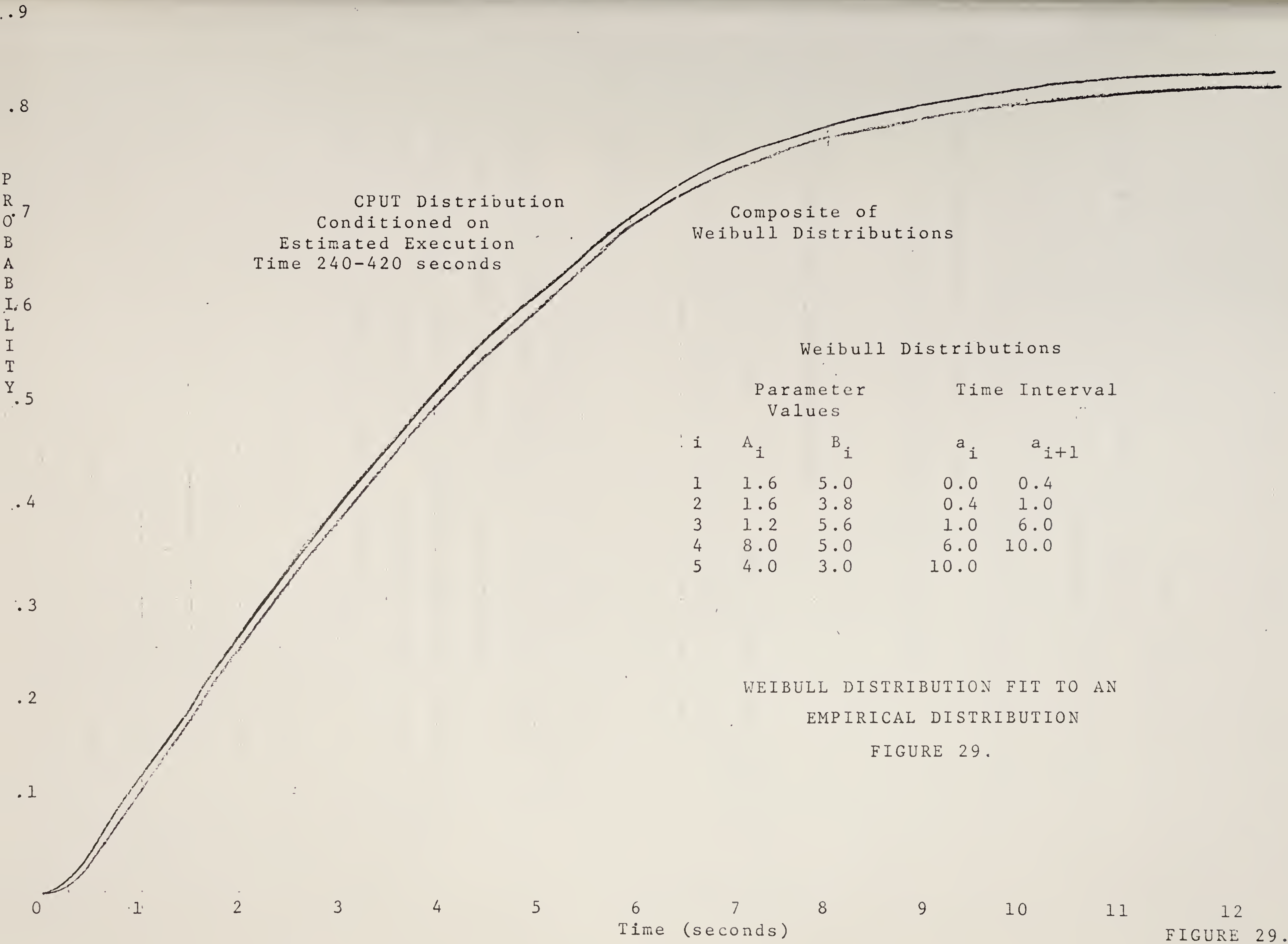
(6) Beta

This distribution was considered but no research with actual data was performed. This is a candidate for future investigation because:

(a) its general shape seems to have a more rapid drop after the maximum value than the gamma distribution, and consequently, perhaps a more extended tail. See Figure 31.

(b) it is finite, which is a characteristic of empirical distributions.

(7) Convolution of Other Distributions



Weibull Distributions

	Parameter Values		Time Interval	
i	A_i	B_i	a_i	a_{i+1}
1	1.6	5.0	0.0	0.4
2	1.6	3.8	0.4	1.0
3	1.2	5.6	1.0	6.0
4	8.0	5.0	6.0	10.0
5	4.0	3.0	10.0	

WEIBULL DISTRIBUTION FIT TO AN EMPIRICAL DISTRIBUTION

FIGURE 29.

FIGURE 29.

It is possible that the empirical distributions are mixtures of other distributions, such as

(a) convolution of an exponential distribution and a hyperexponential distribution, and

(b) convolution of an Erlang-k distribution and a hyperexponential distribution.

No research was performed in this area.

Distribution Characterizations

The mean residual life and rank values were calculated for most of the empirical distributions.

Results

Both the mean residual life and rank are generally increasing with some intervals of decreasing values. The values for mean residual life are consistently and significantly higher than those for rank, at corresponding points of the distributions. Figures 30 and 31 give typical and unusual results for each function.

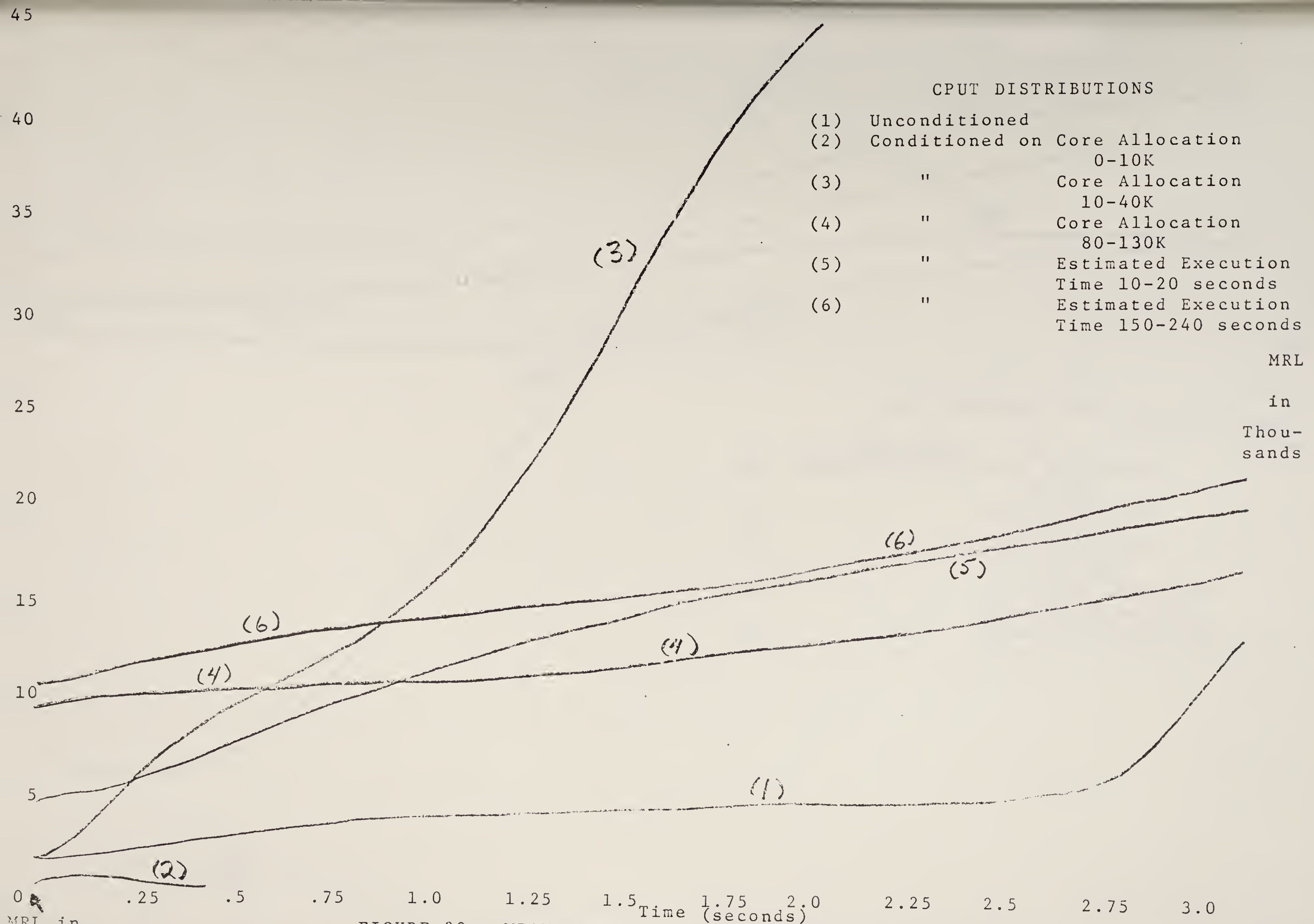
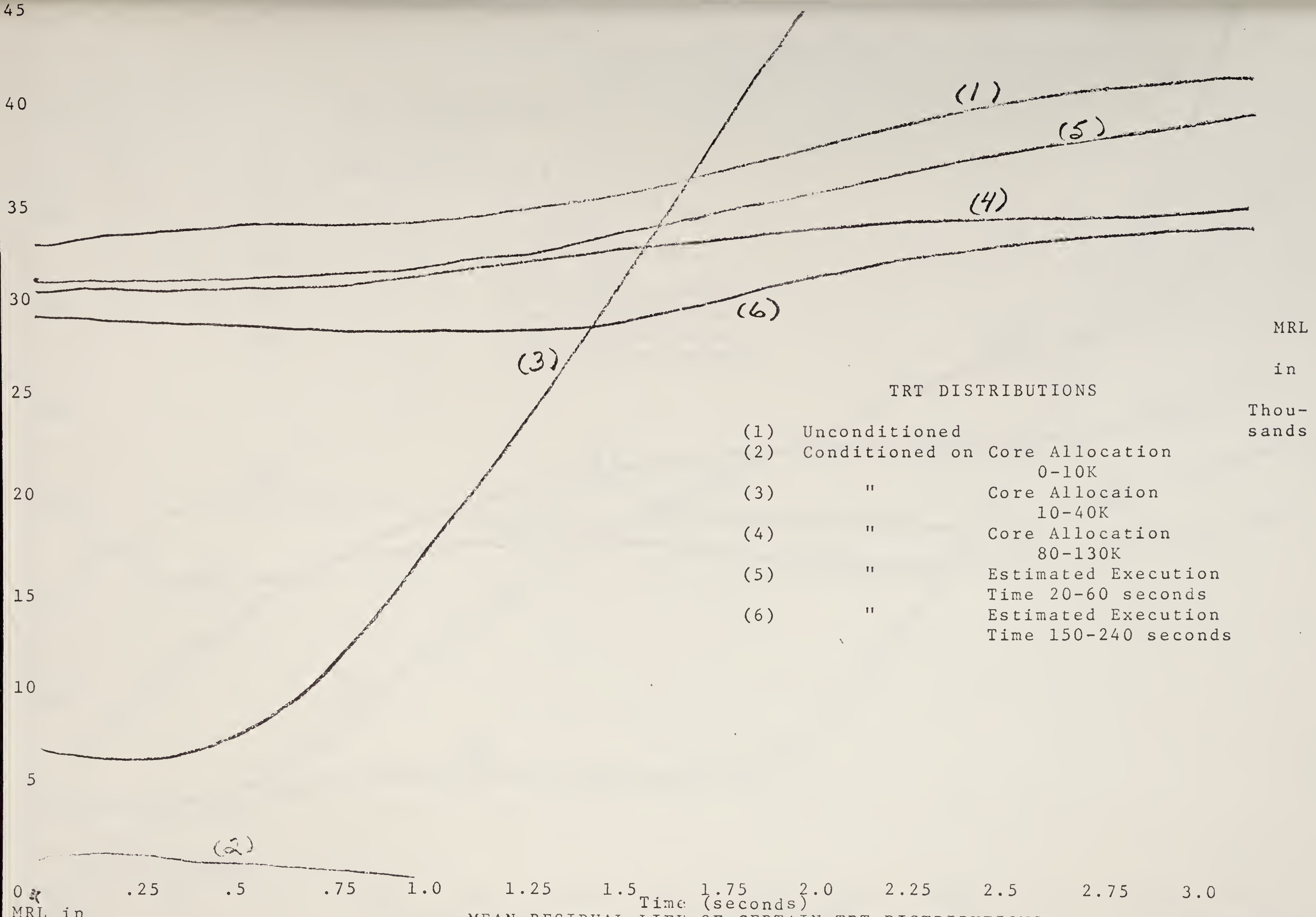


FIGURE 30. MEAN RESIDUAL LIFE OF CERTAIN CPUT DISTRIBUTIONS Page 1 of 2



MEAN RESIDUAL LIFE OF CERTAIN TRT DISTRIBUTIONS

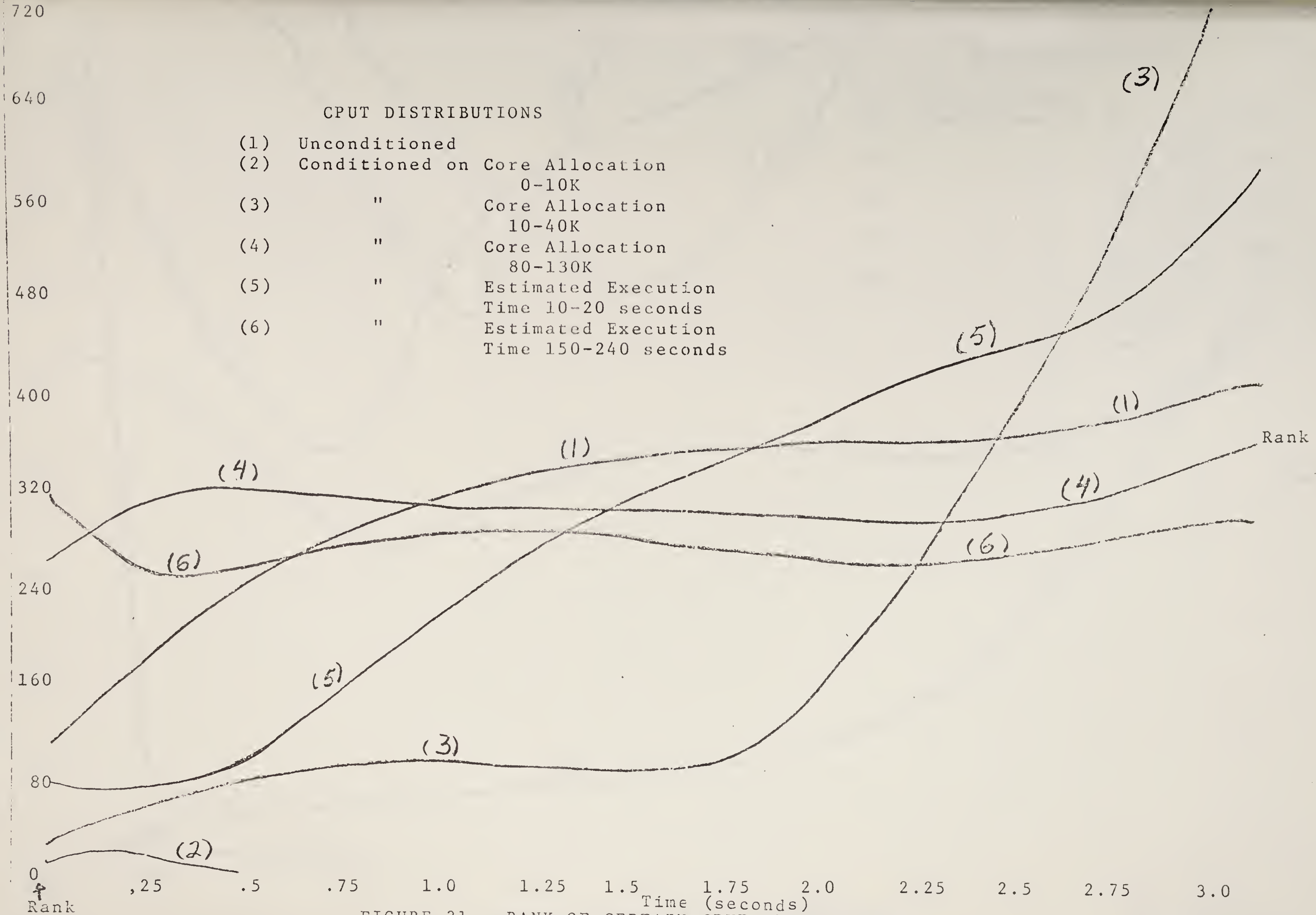
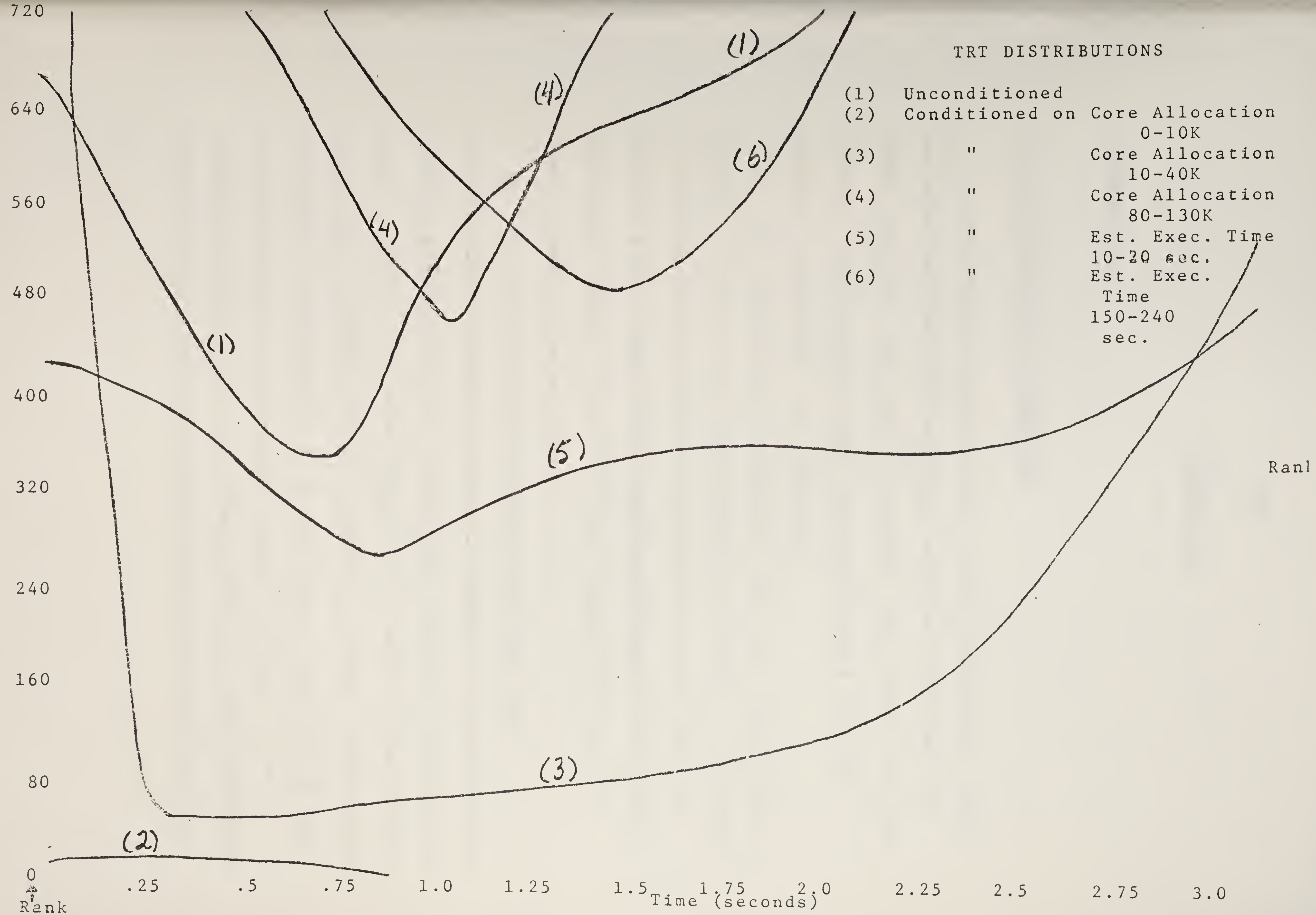


FIGURE 31. RANK OF CERTAIN CPUT DISTRIBUTIONS

Page 1 of 2



RANK OF CERTAIN TRT DISTRIBUTIONS

CHAPTER IV

Conclusions

Best Partitioning Parameters

Our results indicate that certain parameters, alone and when included with another, partition CPU times and total run times into more meaningful distributions than do others. To determine which parameters yield the best results, we consider the following:

(1) The amount of variation from one distribution to another. This is the most important criterion. One factor in determining the amount of variation is the number of classes with a distribution indistinguishable from that of some other class. This is relevant only in the two conditioning parameter case.

(2) The percentage of jobs comprising each distribution. Discrete parameter classes are formed through predetermined criteria. For example, Class is specified by A, B, C, and M. No other adjustment is made to the classes other than combining them if indistinguishable distributions do occur. Consequently, if a very large percentage of the jobs fall within a particular class, the class has more possibility of being a joint distribution of other submerged distributions. With continuous parameters, this factor is

not important due to the amount of interval investigation performed in choosing class boundaries. This factor is of much less importance under two parameter conditioning, particularly so when a discrete parameter is combined with a continuous parameter.

(3) The number of classes with sufficient frequency. The greater the number of classes of a parameter implies the greater the number of distinct distributions in one parameter conditioning. However, in two parameter conditioning, a number of the resulting classes may have low frequency, and thus cannot be considered as a separate distribution.

In the following discussions, parameter references apply to both CPU time and total run time distributions, unless otherwise specified.

Results of One Conditioning Parameter

The amount of variation differs from parameter to parameter. Our results indicate that Core Allocation produces the greatest variation, with Estimated Execution Time performing almost as well. Class and Authorization Code are much less significant in this respect, with Class probably the better of the two. We conjecture that the dependence of Core Allocation on operating system procedures may be responsible in part for the greater variation among distributions for that parameter than among the distributions for Estimated Execution Time.

When considering the evenness of the distribution of jobs over the classes, we find that generally, the continuous parameters tend to be better than the discrete parameters. Estimated Execution Time definitely has the most uniform distribution and Class, the least uniform distribution.

All classes for each parameter have sufficient frequency.

Results of Two Conditioning Parameters

Because of the large number of classes in each two parameter set, it is more difficult to determine which combinations of parameters produce the greater variation in their distributions. Generally, the distributions resulting from CPU time conditioning are much more variable than those resulting from total run time conditioning. Core Allocation and Estimated Execution Time partition CPU time better than other combinations. However, total run time is best partitioned by Core Allocation and Authorization Code, implying that the type of user is of importance when considering I/O times. Class combined with Authorization Code produces the poorest distribution. We note that Core

Allocation combined with Class or Authorization Code conditions significantly better than does Estimated Execution Time with Class or Authorization Code.

Upon considering which parameters produce classes of equal frequency, we find that continuous parameters in combination with discrete parameters have more even distributions than the combination of the discrete parameters.

Once classes with insufficient numbers of jobs in them are eliminated, the number of classes for the combinations of Estimated Execution Time and Authorization Code, and Estimated Execution Time and Core Allocation, are approximately the same. Authorization Code and Core Allocation have fewer classes. Class in combination with any of the other parameters, has roughly the same number of classes.

From our previous discussions, we conclude that the best conditioning parameter is Core Allocation, with Estimated Execution Time and Authorization Code performing almost as well. The poorest is Class. We also conclude that two parameter conditioning yields better results than one parameter conditioning.

One other result of partitioning is worth noting here, that pertaining to the reduction of coefficients in variation. In all conditioning results, the coefficients of variation are reduced from those of the unconditioned distributions in a significant number of classes. Very few were noticeably higher. Thus we can also conclude that partitioning does lower the coefficient of variation significantly in the majority of the partitioned distributions.

These conclusions are relevant only to our operating system and the time period over November and December of 1972. In another system or within another time period, different results may be obtained. Additional parameters could and, if possible, should, be considered in future studies. One that may produce interesting results is a parameter indicating the translator required for the program. Another involves more refinement in the type of user area, i.e., relevant to our data, more distinction as to the type of program submitted in each of the classes of Authorization Code. Perhaps estimates of the actual service times rather than the upper limits of the service times, would prove meaningful.

Distribution Fit

The empirical distributions do not fit satisfactorily to any of the theoretical distributions based on our methods of parameter estimation. Very peaked distributions have little breadth; most distributions have too large a proportion of jobs lying in the tails. We feel that research involving different methods of parameter estimation of the theoretical distributions, may be productive. Investigation of the log-normal, Beta, and convolution of distributions would be worthwhile. One alternative approach to this problem is to fit the functions that characterize the empirical distributions, such as failure rate, mean residual life, or rank.

Implications in the Scheduling Discipline Area

Our results do imply that a priori characteristics of a job can be used to acquire information about the processing time distribution of that job and lower the coefficient of variation of the distribution significantly. Partitioning on more than one parameter is more significant in a multiprogramming environment than in a single task environment, where the effects of varying I/O requirements are felt. These results do have a direct effect on those scheduling disciplines that use this information in their selection rule, i.e. SEPT, SEEPT, and SR.

To effectively use this information, these disciplines must store varying amounts of information. SPRPT and SR must store the greatest amount, as they depend on preemption and thus information gained during the processing of a job. But although these rules may perform better than others with regard to response times and other considerations, preemption and storage costs may decrease significantly the overall benefit of using these scheduling disciplines. Before the full significance of our results can be determined, further research must be directed at studying the effects of these costs, through simulation, through actual system investigation, or through further analysis.

REFERENCES

1. Anderson, H. A. Jr. "An empirical investigation into foreground-background scheduling for an interactive computing system." IBM Research Report RC 3941, Yorktown Heights, N.Y., July 1972.

2. Bryan, G. E. "JOSS: 20,000 hours at a console: a statistical summary." Proc. AFIPS 1967 Fall Joint Comput. Conf., Vol. 31, pp. 769-777.

3. Coffman, E. G., and Kleinrock, L. "Computer scheduling methods and their countermeasures." Proc. AFIPS 1968 Spring Joint Comput. Conf., Vol. 32, pp. 11-21.

4. Coffman, E. G. "Studying multiprogramming computer systems with queueing theory." Datamat. 13, 6 (June 1967), pp. 47-56.

5. Fine, G. H., Jackson, C. W., and McIsaac, P. V. "Dynamic program behavior under paging." Proc. of 22nd Nat. Conf. of ACM, 1967, pp. 55-96.

6. Fuchs, E., and Jackson, P. E. "Estimates of distributions of random variables for certain computer communications traffic models." Comm. of ACM 13, 12 (December 1970), pp. 752-757.

7. Greenberger, M. "The priority problem and computer time-sharing." Manage. Sci. 12, 11 (July 1966), pp. 889-906.

8. Hildebrand, F. B. Introduction to Numerical Analysis. McGraw-Hill, New York, N.Y., 1956.

9. Hillier, F. S., and Lieberman, G. J. Introduction to Operations Research. Holden-Day, San Francisco, California, 1967.

10. Kao, J. "A summary of some new techniques on failure analysis." Proc. 6th Nat. Symp. on Reliability Quality Control in Electronics, 1960, pp. 190-201.

11. Kleinrock, L. "Time-shared systems: a theoretical treatment." J. of ACM 14, 2 (April 1967), pp. 242-261.

12. McIsaac, P.V. "Time-sharing job descriptions for simulation." Report TM-2713, System Development Corporation, 1965.

13. McKinney, J. M. "A survey of analytical time-sharing models." Comput. Surv. 1, 2 (June 1969), pp. 105-116.

14. McNaughton, R. F. "Scheduling with deadlines and loss functions." Manage. Sci. 6, 1 (October 1959), pp. 1-12.

15. Ralston, A. A First Course in Numerical Analysis. McGraw-Hill, New York, N.Y., 1965.

16. Rothkopf, M. H. "Scheduling with random service times." Manage. Sci. 12, 9 (May 1966), pp. 707-713.

17. Schemer, J. E., and Heying, D. W. "Performance modeling and empirical measurements in a system designed for batch and time-sharing users." Proc. AFIPS 1969 Fall Joint Comput. Conf., Vol. 35, pp. 17-26.

18. Scherr, A. L. "Time-sharing measurement." Datamation 12, 4 (April 1966), pp. 22-26.

19. Scherr, A. L. An Analysis of Time-shared Computer Systems. M.I.T. Press, Cambridge, Mass., 1967.

20. Schrage, L. E. "Some queueing models for a time-shared facility." Ph. D. Dissertation, Dept. of Indust. Eng., Cornell Univ., February 1966.

21. Schrage, L. E. "A proof of the optimality of the SRPT discipline." Oper. Res. 16, 3 (May 1968), pp. 687-690.

22. Schrage, L. E. "The modeling of man-machine interactive systems." Report No. 6942, Center for Mathematical Studies in Business and Economics, Univ. of Chicago, September 1969.

23. Schrage, L. E. "Optimal scheduling rules for information processing systems." ICR Quarterly Report No. 26, Institute for Computer Research, Univ. of Chicago, August 1970.

24. Schrage, L. E., and Miller, L. W. "The queue M/G/1 with shortest remaining processing time discipline." Oper. Res. 14, 4 (July 1966), pp. 670-684.

25. Smith, W. E. "Various optimizers for single stage production." *Nav. Res. Logistics Quart.* 3, 1 (March 1956), pp. 59-66.

26. Sevcik, K. C. "The use of service time distributions in scheduling." Report No. 14, Computer Systems Research Group, Univ. of Toronto, May 1972.

27. Tran, T. K. "The response time distributions of some computer scheduling disciplines." M. Sc. Thesis, Dept. of Computer Science, Univ. of Toronto, 1972.

UNIVERSITY OF TORONTO

COMPUTER SYSTEMS RESEARCH GROUP

BIBLIOGRAPHY OF CSRG TECHNICAL REPORTS[+]

CSRG-1 "EMPIRICAL COMPARISON OF LR(k) AND PRECEDENCE PARSERS"
J.J. Horning and W.R. Lalonde, September 1970
[ACM SIGPLAN Notices, November 1970]

CSRG-2 "AN EFFICIENT LALR PARSER GENERATOR"
W.R. Lalonde, February 1971
[M.A.Sc. Thesis, EE 1971]

CSRG-3 "A PROCESSOR GENERATOR SYSTEM"
J.D. Gorrie, February 1971
[M.A.Sc. Thesis, EE 1971]

CSRG-4 "DYLAN USER'S MANUAL"
P.E. Bonzon, March 1971

CSRG-5 "DIAL - A PROGRAMMING SYSTEM FOR INTERACTIVE ALGEBRAIC MANIPULATION"
Alan C.M. Brown and J.J. Horning, March 1971

CSRG-6 "ON DEADLOCK IN COMPUTER SYSTEMS"
Richard C. Holt, April 1971
[Ph.D. Thesis, Dept. of Computer Science,
Cornell University, 1971]

CSRG-7 "THE STAR-RING SYSTEM OF LOOSELY COUPLED DIGITAL DEVICES"
John Neill Thomas Potvin, August 1971
[M.A.Sc. Thesis, EE 1971]

CSRG-8 "FILE ORGANIZATION AND STRUCTURE"
G.M. Stacey, August 1971

CSRG-9 "DESIGN STUDY FOR A TWO-DIMENSIONAL COMPUTER-ASSISTED ANIMATION SYSTEM"
Kenneth B. Evans, January 1972
[M.Sc. Thesis, DCS 1972]

CSRG-10 "HOW A PROGRAMMING LANGUAGE IS USED"
William Gregg Alexander, February 1972
[M.Sc. Thesis, DCS 1971]

CSRG-11 PROJECT SUE STATUS REPORT
J.W. Atwood (ed.), April 1972

CSRG-12 "THREE DIMENSIONAL DATA DISPLAY WITH HIDDEN LINE REMOVAL"
Rupert Bramall, April 1972
[M.Sc. Thesis, DCS 1971]

[+] Abbreviations:
DCS - Department of Computer Science, University of Toronto
EE - Department of Electrical Engineering, University of Toronto

CSRG-13 "A SYNTAX DIRECTED ERROR RECOVERY METHOD"
Lewis R. James, May 1972
[M.Sc. Thesis, DCS 1972]

CSRG-14 "THE USE OF SERVICE TIME DISTRIBUTIONS IN SCHEDULING"
Kenneth C. Sevcik, May 1972
[Ph.D. Thesis, Committee on Information Sciences, University of Chicago, 1971; JACM accepted for publication]

CSRG-15 "PROCESS STRUCTURING"
J.J. Horning and B. Randell, June 1972
[ACM Computing Surveys, March 1973]

CSRG-16 "OPTIMAL PROCESSOR SCHEDULING WHEN SERVICE TIMES ARE HYPEREXPONENTIALLY DISTRIBUTED AND PREEMPTION OVERHEAD IS NOT NEGLIGIBLE"
Kenneth C. Sevcik, June 1972
[Proceedings of the Symposium on Computer-Communication, Networks and Teletraffic, Polytechnic Institute of Brooklyn, 1972]

CSRG-17 "PROGRAMMING LANGUAGE TRANSLATION TECHNIQUES"
W.M. McKeeman, July 1972

CSRG-18 "A COMPARATIVE ANALYSIS OF SEVERAL DISK SCHEDULING ALGORITHMS"
C.J.M. Turnbull, September 1972

CSRG-19 "PROJECT SUE AS A LEARNING EXPERIENCE"
K.C. Sevcik et al, September 1972
[Proceedings AFIPS Fall Joint Computer Conference, v. 41, December 1972]

CSRG-20 "A STUDY OF LANGUAGE DIRECTED COMPUTER DESIGN"
D.B. Wortman, December 1972
[Ph.D. Thesis, Computer Science Department, Stanford University, 1972]

CSRG-21 "AN APL TERMINAL APPROACH TO COMPUTER MAPPING"
P. Kvaternik, December 1972
[M.Sc. Thesis, DCS 1972]

CSRG-22 "AN IMPLEMENTATION LANGUAGE FOR MINICOMPUTERS"
G.G. Kalmar, January 1973
[M.Sc. Thesis, DCS 1972]

CSRG-23 "COMPILER STRUCTURE"
W.M. McKeeman, January 1973
[Proceedings of the USA-Japan Computer Conference, 1972]

CSRG-24 "AN ANNOTATED BIBLIOGRAPHY ON COMPUTER PROGRAM ENGINEERING"
J.D. Gannon (ed.), March 1973

CSRG-25 "THE INVESTIGATION OF SERVICE TIME DISTRIBUTION"
Eleanor A. Lester, April 1973
[M.Sc. Thesis, DCS 1973]